PERFORMANCE ANALYSIS OF A COOPERATIVE SEARCH ALGORITHM FOR MULTIPLE UNMANNED AERIAL VEHICLES UNDER LIMITED COMMUNICATION CONDITIONS

THESIS

Kevin M. Morris, Second Lieutenant, USAF

AFIT/GE/ENG/06-44

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

AFIT/GE/ENG/06-44

PERFORMANCE ANALYSIS OF A COOPERATIVE SEARCH ALGORITHM FOR MULTIPLE UNMANNED AERIAL VEHICLES UNDER LIMITED COMMUNICATION CONDITIONS

THESIS

Presented to the Faculty

Department of Electrical and Computer Engineering

Graduate School of Engineering and Management

Air Force Institute of Technology

Air University

Air Education and Training Command

In Partial Fulfillment of the Requirements for the

Degree of Master of Science in Electrical Engineering

Kevin M. Morris, B.S.E.E.

Second Lieutenant, USAF

March 2006

AFIT/GE/ENG/06-44

PERFORMANCE ANALYSIS OF A COOPERATIVE SEARCH ALGORITHM FOR MULTIPLE UNMANNED AERIAL VEHICLES UNDER LIMITED COMMUNICATION CONDITIONS

Kevin M. Morris, B.S.E.E.

Second Lieutenant, USAF

Approved:

Barry Mullins — 28 Feb 06
Barry E. Mullins, Ph.D. (Chairman) — Date

Michael A. Temple — 28 Feb 06
Michael A. Temple, Ph.D. (Member) — Date

Rusty O. Baldwin — 28 Feb 06
Rusty O. Baldwin, Ph.D. (Member) — Date

Acknowledgments

First and most importantly, I would like to thank my parents, whose encouragement and support helped keep me going even when things became immensely frustrating and stressful. As a result of my experiences and their insight, I was able to learn a good deal about myself. Their assistance in the large amount of proof-reading necessary was also greatly appreciated.

I would like to acknowledge some of the many friends and fellow students that struggled with me through this time, encouraged me to accomplish more, and shared or helped to solve some of the problems that we faced. Specifically, I would like to mention Capt Roger Illari and Capt LeRoy Willemsen, who helped to break up my day and provide momentary escapes from the work. They also helped during the entire process by giving me companions who were experiencing roughly the same difficulties and frustrations as I was.

The wisdom and experience provided by my academic advisor, Dr. Michael Temple, was immensely helpful and greatly appreciated. In addition, I would like to express my appreciation to my thesis advisor, Dr. Barry Mullins, for his guidance and support throughout the course of this thesis effort. Lastly, I would like to thank my sponsors, Dr. Daniel Pack and Lt Col George York, from the Air Force Academy for the support they provided in this endeavor as well as the Air Force Communications Agency for the funding provided.

Kevin M. Morris

Table of Contents

Page

Page

List of Figures

Figure Page

List of Tables

Table Page

AFIT/GE/ENG/06-44

Abstract

This research investigates the impacts of realistic wireless communications upon a group of unmanned aerial vehicles (UAVs) utilizing a distributed search algorithm. The UAVs are used to survey an area for mobile targets and require communication to cooperatively locate the targets. The mobile targets do not continually radiate energy, which exacerbates the search effort; a UAV could fly directly over a target and not detect it. A simulation of cooperative UAVs is implemented using the OPNET Modeler network simulation tool.

The search performance of a group of UAVs is observed when (1) communication range, (2) data rate, and (3) the number of UAVs are varied. The performance is evaluated based on the total time it takes for the UAVs to completely detect all targets in a given search area, the number of times internal areas are scanned, as well as the amount of communication throughput achieved, network traffic generated, network latency, and number of network collisions. The results indicate that the number of UAVs was found to have the greatest impact on the group's ability to search an area, implying the data shared between UAVs provides little benefit to the search algorithm. In addition it is found that a network with a 100 Kbps or faster data rate should allow for minimal congestion and for a large degree of scalability. The findings demonstrate that the proposed four-stage search algorithm should operate reasonably well under realistic conditions.

PERFORMANCE ANALYSIS OF A COOPERATIVE SEARCH ALGORITHM FOR MULTIPLE UNMANNED AERIAL VEHICLES UNDER LIMITED COMMUNICATION CONDITIONS

I. Introduction

In one of Aesop's fables, the value of teamwork is illustrated by the fact that it is difficult to break a group of sticks that are bound together, yet it is a simple task to break them individually once they are no longer tied together. The moral of this story is that many people working together in a team can accomplish more than each of them alone. Processors and computers, carefully programmed, can exploit this same synergy to accomplish useful tasks. However, poor communication between individuals can quickly eliminate the synergy created by working together, similar to unbinding Aesop's sticks. The aim of this research is to investigate the impacts realistic, limited communication has on unmanned aerial vehicles (UAVs) using a distributed algorithm to cooperatively search an area for mobile ground targets.

1.1 Motivation

The Department of Defense is currently interested in the employment of UAVs for many different tasks and views UAVs as a way to transform the military [Gar02] and [Sam03]. Over one billion dollars is budgeted in fiscal year 2006 for the procurement and development of UAV platforms for the U.S. Air Force alone [Jon05]. UAVs are advantageous since they can be "parked" such that they can continually survey an area of interest. They can be used in missions where the risk to human life is great [Gar02].

There are also many future applications of UAVs that will be beneficial. One of these applications takes advantage of teamwork in cooperative UAV systems known as sensor swarms. A sensor swarm consists of many UAVs working together to search an area. A sensor swarm may be homogenous or may include other types of UAVs such as unmanned combat aerial vehicles (UCAVs) that can engage and destroy threats once located and identified by the sensing UAVs. Intelligence-gathering UAVs are expendable and free pilots and larger aircraft for other tasks. Furthermore, cooperative UAVs can autonomously survey larger areas more quickly and have more processing power, redundancy, and the potential for greater synergistic effects than individual UAVs [ViR04].

1.2 Overview and Goals

Pack and Mullins [PaM03] identify a set of universal search algorithm rules for swarms. In their study the search rules are developed for ground based robots. The number of robots that can be supported by the communication network is a topic of interest in their paper. They predict that at most 118 individual robots are able to communicate with one another based upon their stated parameters. Though the study discusses a communication format, the experiments performed appear to only test the rule set.

Related studies [PYT05] and [YPH06] consider a four-stage search algorithm to mitigate the problems caused by mobile, intermittent targets. Since this algorithm is distributed it requires communication amongst the UAVs to cooperatively locate targets and send target coordinates to another platform such as a UCAV. These studies adapt the rule set from [PaM03] for UAVs and include them in the global search stage (stage 1); however, the focus of these studies is on the target localization stage (stage 3) or tradeoffs between the global search stage and the localization stage through group formations. That is, communication between UAVs was not considered. It was assumed each node had the same knowledge of the search space and targets being located, which implies that each UAV could communicate reliably and instantaneously with all other UAVs.

The primary goal of this study is to investigate the impact communication limitations have on the search algorithm in the aforementioned studies. This particular search algorithm is investigated because the sponsors of this work happen to be the researchers in the studies discussed. The results of this study will add an important component to these efforts.

1.3 Thesis Layout

This chapter introduces the topic of study and provides motivation for the research. Chapter II reviews fundamental concepts in this research as well as discusses recent related studies. Chapter III presents the methodology used to carry out the experiments. Chapter IV provides discussion and analysis of the experimental results. Chapter V draws conclusions about the results and offers areas for future research.

II. Literature Review

2.1 Introduction

This chapter presents the fundamental concepts in swarming, Unmanned Aerial Vehicles (UAVs), and wireless networks including recent research. Section 2.2 contains a discussion of swarms, highlighting general concepts important to the field of study. Section 2.3 enumerates some general properties of UAVs, which are the basic swarm elements for this study. Section 2.4 presents relevant current topics in Mobile Ad Hoc Networks (MANETs), which can be viewed as an abstraction of the communication links between elements of a swarm. Section 2.5 briefly touches on OPNET Modeler, a network simulation tool. Section 2.6 narrows the scope of the discussion and focuses on closely related applications and research.

2.2 Swarms

A swarm, in this context, is defined at the most basic level as a group of individual entities generally in motion and potentially working together to perform a task [MeW06]. Individual members of a swarm do not have to be homogeneous, but often are. Many examples of swarm animals exist in nature such as bees, ants, and schooling fish to name a few. In fact, it is these natural swarms that have become subjects of study and inspiration for creating and optimizing man-made swarms and networks [BDT99].

It is difficult to discuss swarms without introducing the concept of emergence. Emergence or emergent behavior refers to the phenomenon where an interesting or beneficial overarching group behavior arises from seemingly simple rules and interactions between individuals [Wik06]. Emergence is also commonly referred to as swarm intelligence, as the individual members are not necessarily aware of the behavior of the group as a whole and are governed by their own simple rules [KEM01]. For example, schools of certain species of fish form very interesting patterns when avoiding predators. The instinctive behavior (or programming) of each individual fish causes it to react in a predictable manner. For instance, some fish selfishly try to hide behind their neighbors. As each fish does its best to hide, the entire school produces a chaotic flow that can confuse potential predators. In other instances, schools have been known to swim in a circular pattern and take on a cyclonic shape as seen in Figure 1 [PVG02].

Figure 1. Cyclonic Emergent Fish School Pattern [PaE99]

Stigmergy is another important swarm concept that is readily observed in ant swarms. Stigmergy is defined as communication through the environment [KEM01]. Often, stigmergy is achieved using pheromones to communicate information or influence individual and group behaviors. Pheromones have three important properties. First is that a pheromone deposited at a location by any individual aggregates with the pheromones deposited by other individuals. Second, pheromones evaporate over time and lastly pheromones diffuse to surrounding areas. Often pheromone diffusion is referred to as gossiping when used in the context of MANETs and wireless sensor networks (WSNs) [PPO02].

The formation of ant trails from the colony to a food source and back is the most often cited example of the use of pheromones. The emergent property evident in this example is that ants, when presented with multiple paths to a food source, will choose the shortest of the paths since the pheromones deposited on the longer paths will evaporate faster and aggregate slower than those on the shorter paths. Over time, the ants are more likely to choose the shorter path. However, this does not necessarily mean that the ants will use the absolute shortest path as it may not have been one of the original paths discovered [BDT99].

2.3 Unmanned Aerial Vehicles (UAVs)

During and after the Vietnam War, robotic aircraft were referred to as remotely piloted vehicles (RPVs). The current term, Unmanned Aerial Vehicle (UAV), became the generally accepted term for robotic aircraft in the late 1990's [New04]. The term UAV is defined by the Department of Defense (DoD) as:

> *A powered, aerial vehicle that does not carry a human operator, uses aerodynamic forces to provide vehicle lift, can fly autonomously or be piloted remotely, can be expendable or recoverable, and can carry a lethal or nonlethal payload. Ballistic or semiballistic vehicles, cruise missiles, and artillery projectiles are not considered unmanned aerial vehicles.* [DoD01]

There are currently many different acronyms associated with UAVs that relate to specific functions or specialization. For example, the term Unmanned Combat Aerial Vehicle (UCAV) refers to UAVs whose primary role is for attack as UAV has become synonymous with surveillance and reconnaissance missions. Small UAVs known as micro-UAVs (MAV) are assumed to be unmanned due to their size and thus that part of the acronym is dropped [New04].

Table 1 shows the characteristics of a variety of the UAVs used by the United States Armed Forces. The Black Widow is a disk-shaped air vehicle and is considered a MAV. The sizes and capabilities of UAVs vary considerably and play a role in the selection of a UAV for a particular mission. Sometimes a desired mission can determine the design of a new UAV [New04].

Table 1. Selected UAVs and Their Characteristics [Jan05]

Name	Wing span	Length	Launching Weight	Ceiling	Cruising Speed	Endurance
Black Widow	150 mm		60 g	Not Listed	23 kts	22 min
Desert Hawk	1.12 m	0.71 m	2.3 kg	305 m	30-50 kts	60-90 min
Dragon Eye	1.14 m	0.91 m	2.49 kg	366 m	35 kts	30-60 min
Global Hawk (RQ-4A)	35.41 m	13.51 m	12,111 kg	>19,810 m	343 kts	35 hours
LOCAAS	1.18 m	0.91 m	45.4 kg	Not Listed	200 kts	30 min
Pathfinder	30.48 m	2.44 m	218 kg	21,340 m	31 kts	14 hours
Predator	16.15 m	8.13 m	1,020 kg	7,620 m	73 kts	> 40 hours
Silver Fox	2.13 m	1.52 m	10 kg	> 305 m	70 kts	10 hours

There is some disagreement on what constitutes a UAV. Though specifically ruled out in the Department of Defense definition, some consider ballistic vehicles or cruise missiles to be expendable “one-way” UAVs. It is argued that there is little difference between a cruise missile that can self-navigate and control its speed and trajectory so as to arrive on target at a designated time and an autonomous craft that performs the same function and carries the payload to the target. Some authors use the term UAV to refer to a flying munition that can sense, navigate, detect a target, and make decisions [LAN03]. Though not stated, the munition might even be recoverable if not used. The deciding factor whether a vehicle is a UAV depends on whether the payload is an integral part of the airframe. If so, the vehicle is usually not considered a UAV [New04].

Another disagreement has to do with the classification of drones. The term drone is commonly associated with manned aircraft that have been later converted to an automated system. An example of this is the QF-4 used as a target drone. Some argue that UAVs are built without any provision for carrying a human operator or that UAVs are controlled from the ground. However, counterexamples such as the U.S. Navy's optionally-piloted Pelican and Soviet remotely-controlled manned interceptors such as Flagons, Foxbats, and Foxhounds yield a general consensus that drones are considered UAVs [New04].

2.4 MANETs

Mobile Ad Hoc Networks (MANETs) is a very descriptive title for the entity being named. A MANET is a network made up of mobile nodes that communicate without a pre-existing infrastructure, such as routers or access points. An example of a MANET would be when a group of students got together to study and each had their own laptop computer with a wireless network card. The students could form an ad hoc (peer-to-peer) network by configuring their computers with the appropriate settings. It is likely nodes would move around from time to time changing route information between nodes and potentially causing some nodes to become disconnected from the network. A collection of airborne UAVs that communicate over a wireless link can also be considered a MANET.

The rest of this section focuses on several topics related to networks in general with emphasis on issues related to MANETs. Section 2.4.1 describes the Open Standards Interconnect (OSI) Network Reference Model, which most networks are based upon. Section 2.4.2 discusses challenges faced by wireless networks and MANETs compared to traditional wired networks. Finally, Section 2.4.3 discusses current standards and technologies relating to MANETs.

2.4.1 OSI Network Reference Model

The International Standards Organization (ISO) developed the OSI Network Reference Model to standardize the way that systems interconnect with one another through a network [PeD03]. The OSI Network Reference Model splits the tasks related to communicating via a network into seven layers shown inside Host A and Host B in Figure 2. Each layer uses the services of the layers below it, thereby higher layers establish virtual connections to the same layer in the destination host. These virtual connections are represented by dashed arrows in Figure 2. All connections between layers are virtual connections with the exception of the physical layer, which exists at the bottom of the layer stack. Consider a computer program at the application layer, which needs to communicate with another host and makes a request for a connection to the layer below it. When the connection is ready, the lower layer signals the application program and the program can now communicate with a remote program on the other host. As far as the application layer program is concerned it is communicating directly with the remote application layer and is oblivious to the details of establishing and maintaining the actual connection.

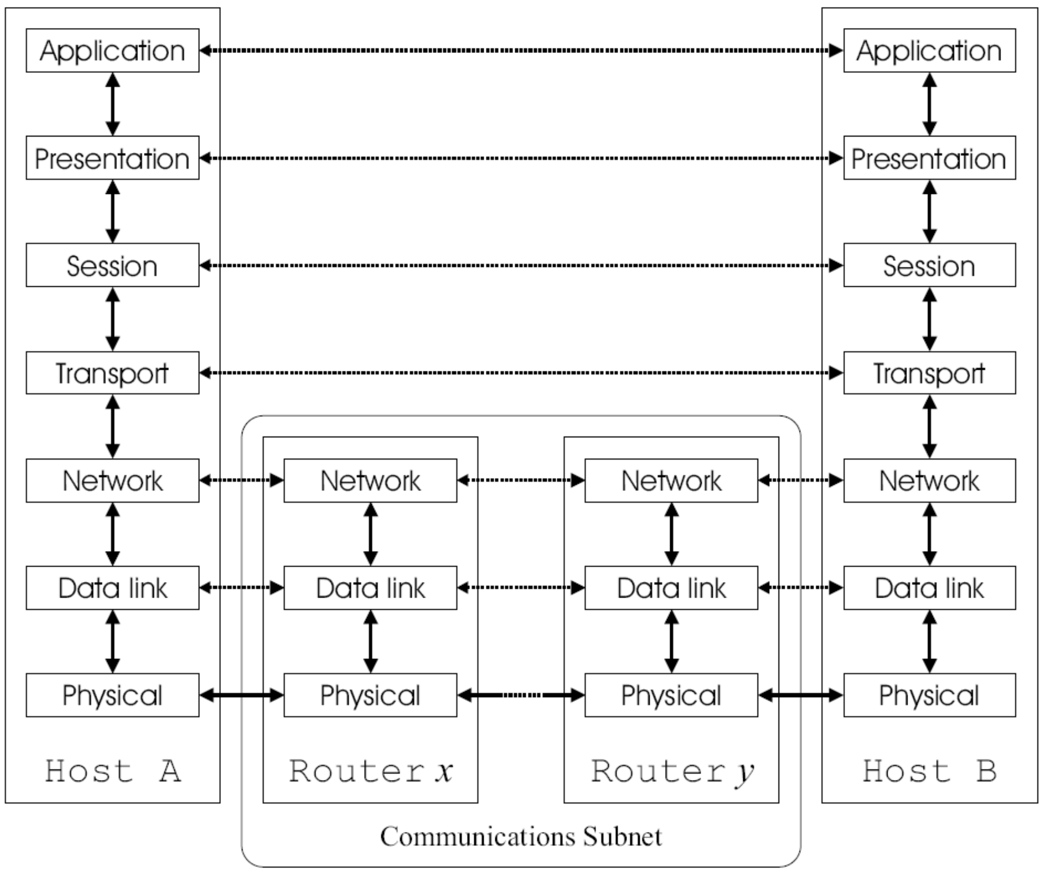

Figure 2. OSI Network Reference Model [Kad03]

Figure 2 shows two routers between the two hosts, however, in reality there could be any number of intermediate routers. It is not necessary for the routers to operate at layers above the network layer. Routers provide connections between nodes on distant networks or that may be using different communication mediums. Routers also provide mechanisms to choose which routes are used during communication.

The abstraction of the layers in the OSI model is also beneficial because of the flexibility and compatibility provided to developers and users. When a small change needs to be made in one layer, that layer can be interchanged without changing the other layers.

Though the OSI model offers flexibility and compatibility, it also incurs more overhead to implement due to the need to manage communication and access between layers. In some implementations, two or more of the layers are combined to increase efficiency. It is fairly common for this to occur between the physical and data link layers [Thu02]. In many wireless systems, system power is constrained and drives the need for greater efficiency to conserve system resources.

2.4.2 Wireless Network Considerations

There are a number of issues in wireless networks that are of much greater importance than in traditional wired networks. Most notably, bandwidth is a fraction of that in wired networks and bit error rates (BER) are significantly greater in wireless networks. Bandwidth constraints arise from the fact that there is only a finite amount of available spectrum with which to transmit and many systems are competing for these resources. These competing systems contribute to the increased BER as does the increased presence of noise and other channel effects. A higher BER causes packets to become corrupted and subsequently discarded in wireless networks. The transmission control protocol (TCP) uses the number of lost packets to determine how fast a node sends information. Packet loss causes TCP to reduce the rate of transmission. In a wired network, transmissions are nearly error free and most data loss is due to network congestion. However, in a wireless network packet loss is much more often due to the BER. When TCP is used in wireless networks, the loss of packets due to corruption causes an unnecessary reduction in the transmission rate and a less efficient use of the available bandwidth [Kad03].

The amount of available system power is often a limiting resource for wireless nodes. Because nodes are often designed to work without the need for restrictive wires, a portable power source is required. Power sources such as batteries eventually expend their energy stores. In consumer products such as cell phones, recharging the nodes is not a problem; however, in some networks, recharging may not be a readily available option and to keep the nodes working over long periods of time may require nodes to conserve power.

The mobility of nodes in a wireless network is also an issue. In a wired network mobility is considered to be very slow or nonexistent. Physically moving the node without changing where it hooks into the network is not considered mobility as the device is still tethered. As the nodes move back and forth in an untethered sense, the topology of the network can change dramatically. It is the fact that the topology of the network can change based upon the physical movement of the nodes that qualifies the nodes as having mobility. Depending on the speed, relative frequency, and direction a node moves, algorithms optimized for network operations such as routing will likely suffer or may not function at all.

The pattern of node movement is sometimes described by a mobility model. Mobility models are generally divided into two categories: individual mobility models and group mobility models. In an individual mobility model, each node moves independently of any other node. However, in a group mobility model, nodes move in groups, though the groups can move independently of other groups if multiple groups exist. Hybrid mobility models combine an individual mobility model with a group mobility model [CBD02].

The *random waypoint mobility model* is an individual mobility model. When a node reaches a destination it waits a random amount of time before picking another random destination and uniformly distributed random speed. Another individual mobility model, known as the *random direction mobility model*, has mobile nodes pick a new random direction and speed once the node reaches the boundaries of the simulation area. One example of a group mobility model is the column mobility model. In the *column mobility model*, each group of mobile nodes forms a column and moves forward together in the same direction. Another group mobility model is the *reference point group mobility model*, where nodes move along with a mobile reference point. All members within the group have a generally fixed position with respect to the reference point. For example if the reference point moves to the east, all of the nodes in the group move the same distance to the east. This implies nodes somehow communicate or sense the position of the reference point. A hybrid protocol combines the reference point group mobility model with the random direction mobility model such that the boundaries for a group of mobile nodes, individually using the random direction model, acts as a moving "reference point". It has been shown that a mobility model has a significant impact on the performance of communication protocols in a wireless network [CBD02].

2.4.3 Current Standards and Technologies

2.4.3.1 IEEE 802.11

Currently one of the most common standards in wireless networks is the IEEE 802.11 set of standards. The 802.11 standard specifies both the physical and data link layers of the OSI reference model. The basic layout of the 802.11 standard is shown in Figure 3. Like the IEEE 802.3 protocol which defines the standards for traditional Ethernet networks, the 802.11 standards use the IEEE 802.2 standard, which defines the logical link control layer. The logical link control (LLC) layer exists above the data link layer. The 802.11 Media Access Control (MAC) layer is beneath the LLC and both of these layers comprise the data link layer from the OSI model. The physical layer is comprised of two sublayers called the physical layer convergence procedure (PLCP) sublayer and the physical medium dependent (PMD) layer [Thu02].

The PMD layer is tightly bound to the particular physical layer used for communication while the PLCP layer masks the differences in physical layer options from the MAC layer. The physical layer provides three basic services to the MAC layer: transmission, reception, and carrier sensing (for Carrier Sense Multiple Access (CSMA) described later). As Figure 3 indicates, IEEE 802.11 provides several options for a physical layer. In the original 802.11 standard, three options were available, a Direct Sequence Spread Spectrum (DSSS) option, a Frequency Hopped Spread Spectrum

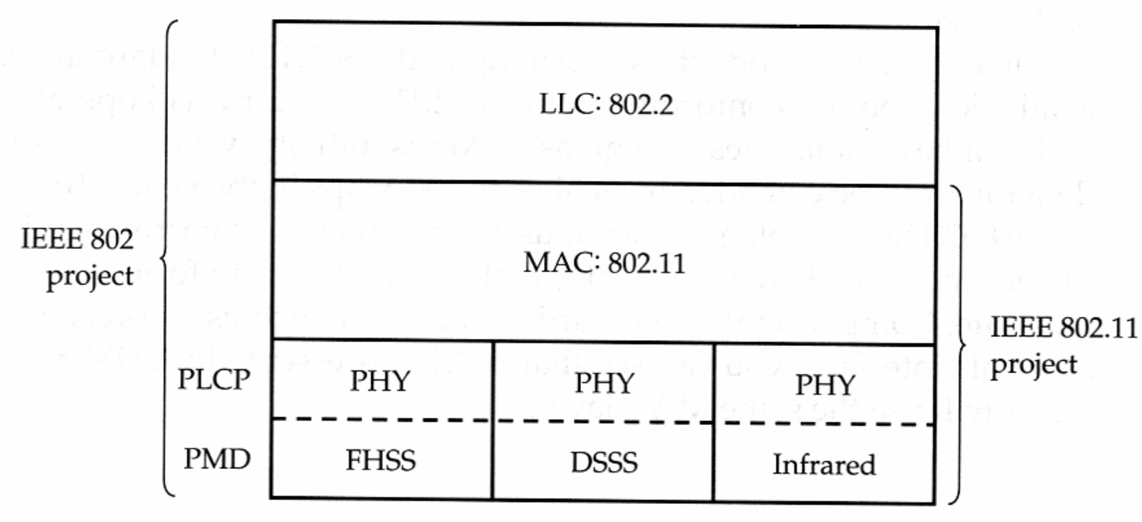

Figure 3. 802.11 Basic Layout [Thu02]

(FHSS) option and an Infrared option. The data rates available in the original specification were 1 and 2 Mbps. The 802.11b amendment allowed for faster data rates of 5.5 and 11 Mbps. The 802.11a amendment specified a data rate of 54 Mbps using Orthogonal Frequency Division Multiplexing (OFDM) as the physical layer option. The frequencies used by 802.11 networks are in the unlicensed industrial, scientific, and medical (ISM) 2.4 GHz band with the exception of 802.11a, which operates at 5 GHz. The 802.11g amendment uses the OFDM option introduced in 802.11a in the 2.4 GHz band. However, 802.11g also supports backward compatibility with 802.11b [Thu02].

Though 802.11 is sometimes referred to as “wireless ethernet”, there are some distinct differences between 802.11 and 802.3. One of the differences is that the 802.11 MAC layer uses Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA), whereas 802.3 uses Carrier Sense Multiple Access with Collision Detection (CSMA/CD). In a Carrier Sense Multiple Access (CSMA) scheme a node that has data to send will first sense the network medium to determine whether or not the medium is in use by another node. If so, the node will delay sending the information until the network media is free. Simply put, the host listens and waits for a gap before transmitting. However, it is possible that detection of transmissions between two distant nodes can be delayed. Should two nodes both sense the medium is free and start to transmit at about the same time then their transmissions are said to collide, which causes both to be received incorrectly.

CSMA/CD adds the additional feature of collision detection to the CSMA scheme. Collision detection, as the name implies, allows for collisions to be detected. The transmitting nodes using the collision detection feature listen to the medium while sending. Should the signal on the medium be different than the signal being sent by the transmitting node, the transmitting node interprets this to mean that a collision has occurred. Once a collision is detected, the transmitting nodes stop transmitting and wait a random amount of time before attempting to resend their information [Thu02].

Collision detection relies on the ability of a node to both transmit and receive at the same time. In the case of a wireless network, antennas are employed to place the signal on the medium. In most cases, a single antenna both transmits and receives signals, which cannot be done at the same time. Even if two antennas were used, one for receive and one for transmit, the signal from the transmitting antenna would greatly overpower any signal received at the receiving antenna during transmission making collision detection very difficult to accomplish. For this reason, 802.11 does not use CSMA/CD and instead employs CSMA/CA [Thu02].

Rather than detecting collisions, CSMA/CA tries to avoid them. Carrier sensing is still used to determine whether or not the medium is free; however, in IEEE 802.11's avoidance scheme, the medium must be idle for at least the Distributed Inter Frame Space (DIFS) before a node transmits. Once a DIFS period has elapsed, each node desiring to transmit will wait an additional random time before transmitting. If the medium has been idle for a long period of time (i.e., no nodes have information to send) a node can send its packet immediately and does not need to wait an additional random time before transmitting. Assuming no collisions occurred and the packet is correctly received, the receiving node will wait a Short Inter Frame Spacing (SIFS) period and transmit an acknowledgement (ACK) back to the sender. The SIFS is shorter than the DIFS, so that the ACK can be sent before any packets that must wait the DIFS time before being transmitted. Priority is given to the ACK as it only has to wait a short time before being sent. The acknowledgement is required in a CSMA/CA scheme to let the sender know that its information was correctly received as the sender is unable to detect collisions [Thu02].

Optionally, 802.11 can employ Request to Send (RTS) and Clear to Send (CTS) signals along with CSMA/CA. The use of the RTS/CTS signals helps to prevent issues related to the hidden node problem. Consider a network that consists of three nodes A, B, and C. Suppose A and B can communicate, B and C can communicate, but nodes A and C cannot communicate, as depicted in Figure 4. The dashed lines indicate the communication range of the node at the center of the circle. A node is capable of communicating with another node if the node is within range (the circle). The reason A and C are unable to communicate could be due to either great distance or an obstacle such as a mountain. Should A and C both try to transmit to B simultaneously, neither would be able to sense the other's transmission and neither packet would successfully arrive at B. If instead, one of the nodes was to send a RTS signal to B, then all nodes within the sender's transmission range have knowledge of the coming transmission. Upon successfully receiving the RTS, node B would send a CTS (if the medium was idle), which informs nodes in range of B of the coming transmission. Finally, the transmission itself and the ACK would be sent [Thu02].

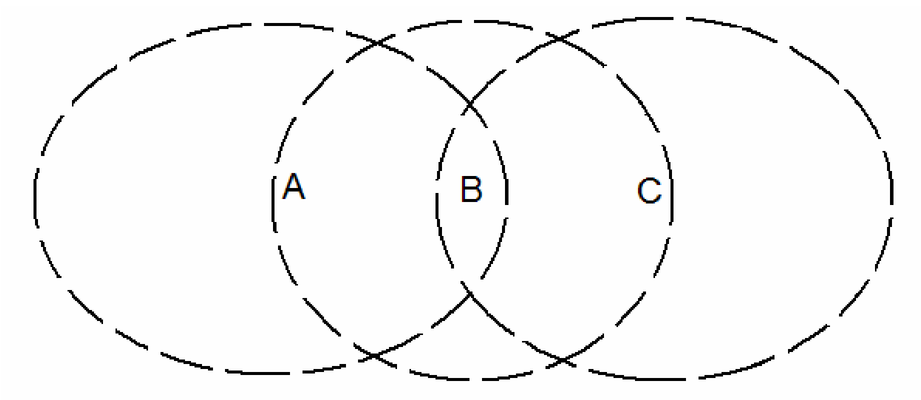

Figure 4. Generic Network Layout to Illustrate the Hidden Node Problem

Broadcast packets cannot easily use the RTS/CTS scheme since a single RTS would potentially cause many CTS packets to be generated. Many of these CTS packets would likely collide making it difficult to determine if the communication medium has been properly reserved. A similar problem occurs if ACK packets are sent in response to broadcast packets as well. Many ACKs would collide, making it difficult to tell if the data was reliably sent to all destinations. For this reason, the 802.11 standard does not use RTS, CTS, or ACK packets in conjunction with broadcast data packets. A broadcast data packet can be sent if the medium has been idle for the length of a DIFS period.

Though the RTS/CTS packets solve the hidden node problem they introduce what is known as the exposed node problem. Suppose an additional node D is added to the network of nodes just discussed. D is only within C's communication range. Figure 5 shows a possible network configuration that illustrates this situation. If B had traffic for A and a RTS/CTS pair was exchanged between A and B, node C would wait for A and B to finish. However, if C had information to send to D it could transmit that information and not interfere with reception at A nor would B's transmission interfere with C's. In practice, this transmission opportunity is not utilized and the network is not as efficient as it could be.

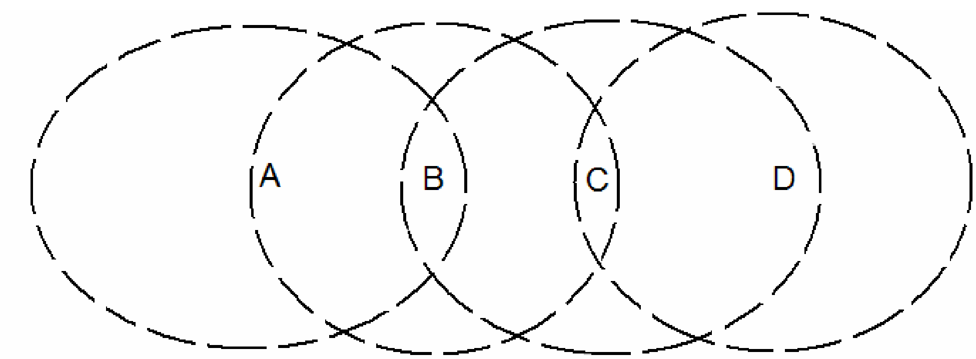

Figure 5. Generic Network Layout to Illustrate the Exposed Node Problem

The 802.11 standard supports two different modes known as "ad hoc" mode when nodes communicate directly with each other and "infrastructure" mode when an access point or router is present. In infrastructure mode, individual nodes do not communicate directly with each other. Instead the nodes communicate through the access point or router present.

The 802.11 standard introduces additional terminology describing how multiple computers can be collectively configured. The range or area covered by an access point is called a Basic Service Set (BSS) when the access point is attached to a distribution medium connecting it to a larger network. The distribution medium can be a traditional wired network or could be a wireless link. An access point or router not connected to a distribution medium or other network resources forms an Independent Basic Service Set (IBSS). An IBSS is not considered an ad hoc network and vice-versa. An Extended Service Set (ESS) is formed by multiple BSSs connected by the distribution medium. ESSs have more than one access point and allow nodes to seamlessly switch between access points. To the LLC layer, an ESS looks like a single large network. Each BSS has a BSS identifier (BSSID), and analogously an ESS has an ESS identifier (ESSID). In 802.11 networks the BSSID is often the 48-bit MAC address of the access point and the ESSID is the network name [Thu02].

2.4.3.2 Routing in a Wireless Environment

Routing can be described as "the process in which a route from a source to a destination node is identified and is achieved" [RoT99]. Some goals in finding a route include distributed computation of routes, minimizing the number of nodes involved, avoiding and eliminating invalid routes, converging on optimal routes, limiting the number of broadcasts, and having backup routes. Routing protocols are often classified as either proactive or reactive and either table-driven or source-initiated (demand-driven).

A proactive protocol actively attempts to determine routes so that routes are ready when needed while a reactive protocol searches for a route when it is needed. Generally, table-driven protocols are proactive and reactive protocols are source-initiated. Examples of table-driven protocols are Destination-Sequenced Distance-Vector Routing (DSDV), Clusterhead Gateway Switch Routing (CGSR) and The Wireless Routing Protocol (WRP). Source-initiated protocols include Ad Hoc On-demand Distance Vector Routing (AODV), Dynamic Source Routing (DSR), Temporally Ordered Routing Algorithm (TORA), and Associativity-Based Routing (ABR) [Ric05].

Flooding is a very simple routing strategy in which a node sends a packet to every node within it's communication range. Each of those nodes, in turn, sends the packet onto every node within the sending node's communication range. Assuming the destination is connected to the network, the packet will traverse the network in the shortest possible time. Because the number of packets exponentially increases over time, there is potential for a large amount of traffic to be generated that does not produce useful results. If several nodes used flooding to send messages, the amount of traffic on the network could significantly lower the ability of the network to send any new traffic. Some routing protocols, such as Open Shortest Path First (OSPF), use flooding to gather information about the network to determine a route. Many routing protocols limit the number of retransmissions and traffic generated to mitigate the primary weakness of flooding [Thu02].

Currently, two of the most discussed and studied routing protocols related to MANETs are the Ad Hoc On-demand Distance Vector (AODV) routing protocol and the

Dynamic Source Routing (DSR) protocol. Both are source-initiated protocols, which are attractive in MANETs since they consume less power in networks that have periods where little or no traffic is sent.

In AODV, when a node has a message to send to some destination for which it does not currently have a valid route, it enters a phase known as path discovery so that an appropriate route can be found. A valid route to a destination may not be known because the destination node is unknown to the source, the route to the destination has expired, or the route has been marked invalid. To discover a route a source node broadcasts a route request (RREQ) to its neighbors, who forward the message until the destination node is found or a node who has a valid route to the destination responds. If the source node moves, it can restart its path discovery protocol to find a path to the destination. If an intermediate node moves, the node just before it in the path will send a failure notification back to the source (and intermediate nodes between the source) [RoT99].

DSR is similar to AODV in that it uses the same basic process to route; however, DSR incorporates several optimizations to save time, memory, and the amount of traffic sent. AODV requires each node to keep its own routing tables while DSR puts the series of network addresses needed to route the packet into the packet itself. This puts the burden of knowing the entire route on the sender, but eliminates the need for intermediate nodes to have up to date routing tables. DSR also takes advantage of the list of addresses included in the packet by allowing each node to eavesdrop on the route being used and thus gain knowledge of the routes other nodes have [Wik05].

2.5 OPNET Modeler

OPNET Technologies, Inc. is a provider of network modeling and simulation software with a number of different tools. The OPNET Modeler software package is an event-based network simulator that is primarily based upon graphical user input. This tool comes with many standard models for commonly used network applications ranging across both wired and wireless technologies. Protocols such as IEEE 802.11, AODV, and DSR are supplied with OPNET Modeler version 10.5 and can be readily implemented in a simulation project. OPNET Modeler does allow for the creation of new models by either modifying existing models or building new models.

At the lowest level OPNET Modeler uses Finite State Machine (FSM) models, which are a collection of states and transitions linked together. Procedures are written in C that can execute when a particular state is entered or exited or during a transition. OPNET Modeler provides extensive libraries of C code to simplify many of the actions that may need to be performed, such as sending and receiving packets, updating a model's attributes, or creating and updating simulation statistics. OPNET products are currently the standard for Department of Defense network simulations.

2.6 Related Research

The Department of Defense (DoD) is very interested in the application of UAV swarm technology. There are two broad areas of interest in UAV swarm deployment, which are sensor swarms and communications swarms. Communication swarms aim to extend the range of current communications technology by providing a low-cost redundant network [ViR04]. Satellites are expensive and not always available so having a swarm of communications relays is much more flexible, cost less, and is more robust. The sensor swarm application employs many UAVs working together to search an area [ViR04]. Additionally, the sensor swarm may include other UAVs or UCAVs that can engage and destroy potential threats once located and identified by the sensing UAVs. The United States is currently finding that more and more often it is engaged in asymmetric warfare where the enemy is likely to seek out urban environments to counteract the United States' technological advantages [Gle99]. Satellite surveillance is not as effective in urban environments, and the United States is reluctant to use weapons that may harm civilian bystanders. The sensor swarm is the proposed solution to searching both urban areas and more traditional rural areas, because sensor swarms are cheaper than satellites and much more flexible [ViR04]. This research is intended to focus on the sensor swarm application rather than the communication swarm application.

Parunak, Purcell, and O'Connell proposed and simulated a system that makes use of unattended ground-based nodes to create an environment for swarming UAVs (and/or unmanned ground vehicles) that uses digital pheromones to mimic natural pheromones [PPO02]. In this particular study, two separate software agents, known as place agents (residing on the ground-based nodes) and walker agents (residing on the mobile nodes), interact to direct the UAVs. The results of the study indicate that larger swarms perform better than smaller swarms and that most swarm systems outperform the baseline that came from a previous Joint Forces Command (JFCOM) study, Unified Vision 00 [PPO02]. It is also noted that better performance was observed when the number of sensors deployed is increased as opposed to changing the swarming algorithm.

Lua, Altenburg, and Nygard present a swarm of airborne autonomous munitions that make coordinated multi-point attacks on targets found within a search space [LAN03]. Each airborne weapon lacks global knowledge of the battlefield and relies on sensors and limited communication with other airborne weapons. Coordination between weapons is performed by using small circular tracks at particular distances from the target. Once a number of weapons are assembled, the weapons break their tracks, position themselves around the target, and then attack it. Conclusions drawn in the study state that the control strategy used can be effective, robust, and scalable. Comparisons between this and global-level control show that the model requires modest communication resources.

A UAV line formation is proposed in [ViR04]. In this study the targets were not considered to be mobile; because of this, the swarm is able to consider an area cleared once searched and does not necessarily need to return to that area. The line formation is proposed to maximize area searched in a minimum amount of time, keep UAVs from colliding in mid-air, to ease reformation of the line in case a UAV is lost or destroyed, and help keep communication between UAVs simple and reliable. The study analyzes the performance tradeoff between the number of UAVs used and the amount of time required for the search. A method for selecting the minimum number of UAVs needed to meet a particular search time within some guarantees is also presented.

Pack and Mullins, in an effort to create a universal search algorithm for swarms, identify a set of rules [PaM03]. The rule set developed consists of four rules as applied to ground-based robots which are, in order: (1) Given a current position, identify the next closest position as the place to visit, (2) If multiple positions are equally close, select the one that lies farthest from the position of the closest robot, (3) If multiple positions are equally far from the other robot, pick the one that lies along the direction just traveled, and (4) If multiple nodes satisfy the first three conditions, pick the next position randomly. An item of concern presented in the study relates to protocol scalability. Given a simple packet structure with a size of 80 bits (containing a single robot position), a 9600 bit per second data rate, a DIFS time of 128 μs, and the assumption that each robot needs to send its position each second, the best case scenario suggests that a swarm of 118 individuals that could all receive from each other would be able to communicate. It is likely that far fewer will actually be able to communicate. The proposed communication scheme in [PaM03] is a modified version of CSMA/CA where the packets are broadcast, acknowledgements are not sent, and retransmissions are not attempted due to the fact that older data is perishable and not every packet is needed. Though the study discusses the communication format, it appears that only the rules were experimentally tested. It was assumed that each node had a global knowledge of the search space, which implies that each node could communicate perfectly with all other nodes.

In related works [YPH06] and [PYT05], which build off of the previous study in [PaM03], a four-stage search process is defined. These four stages are: 1) global search for targets, 2) approach a detected target, 3) orbit and refine a target location, and 4) local search for a lost mobile target. Stage 3 is the main focus of [PYT05], where sensors of differing quality are investigated in the localization process. The main focus of the work

[YPH06] is the impact of a tradeoff made between stages 1 and 3 by either using individual UAVs or formations of three UAVs. In the case where individual UAVs are searching a space, when a UAV finds a potential target it requests that other UAVs come to assist in the finding the exact location of the target through a form of triangulation. If a UAV's cooperation is requested, the UAV first assesses the cost of helping in triangulation versus continuing the search. If the distance is too great or there are already three UAVs assisting, then the UAV continues to search the area. The reason that three UAVs is of importance has to do with information presented in the paper which shows that for each UAV over three, little additional information is gained when locating the target [YPH06]. This relationship discussed in the study is shown in Figure 6 below. The reasoning behind the use of the UAVs in formation is that the UAVs could already be in an appropriate pattern to more quickly find the exact location of the target. In the experiments performed it was assumed that each UAV had a global knowledge of the status of other UAVs, implying perfect communication between UAVs.

Though important steps have been taken in the work done in [PaM03], [YPH06], and [PYT05], these studies do not include the effects of realistic wireless communication. Currently, complete and accurate information such as UAV positions, the number of UAVs in orbit around a target, and when areas were last searched is available to each UAV. If wireless communication is used it is likely that not all of the information each UAV has available is necessarily complete nor the same as information another UAV has due partially to the fact that communication can be lost or garbled or the network

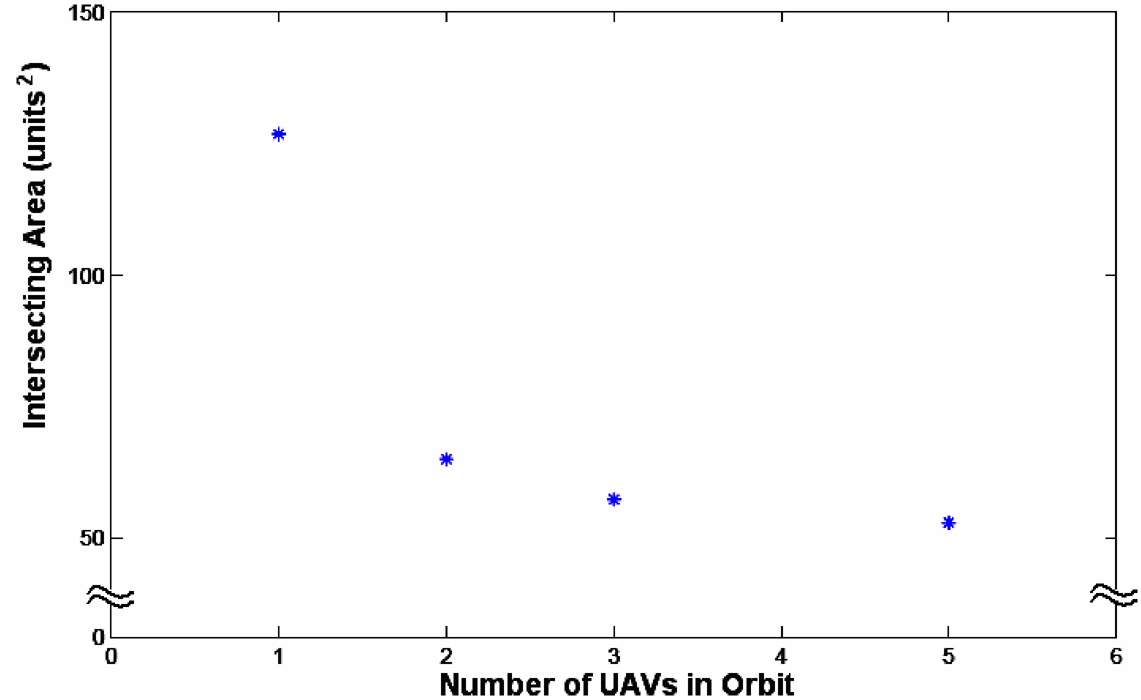

Figure 6. Intersecting Area versus Number of UAVs in Orbit [YPH06]

capacity is insufficient to transfer all of the information. The previous studies do present a communication format suitable for stage 1, but do not present any for later stages. This research adds the effects of realistic communications to determine the impact this has on the search algorithm. However, only the first three stages are investigated in this work. Many of the details of the algorithms presented in [YPH06] and [PYT05] have not been discussed here, but are presented in Appendix A. Also included in Appendix A are additional specifications for implementation-specific details used in this study, such as a communication format for stages 2 and 3.

2.7 Summary

This chapter first introduced the topic of swarms followed by a description of UAVs and a listing of some characteristics of current UAVs. After this was a discussion of MANETs including some of the current issues and topics of interest. A short description of OPNET Modeler was also given. Lastly, related research efforts were presented.

III. Methodology

This chapter provides the methodology used to investigate the impact limited communications conditions have upon the search algorithms developed in [PaM03], [YPH06], and [PYT05]. The necessary information to reproduce the experiment is given here; however, many low-level details and additional protocol information that may be useful are contained in Appendix A.

3.1 Problem Statement

3.1.1 Goals and Hypothesis

The goal of this study is to determine the performance of a UAV sensor swarm engaged in an area search under various communication conditions. These conditions include channel loss and packet collisions. Specifically, the goal investigates the impact of increasing the number of UAVs exchanging packets at various data rates with varying effective communication ranges, thereby determining the number of UAVs that can communicate effectively at a given transmission data rate and communication range.

The number of nodes, data rate, and communication range should impact the time it takes to search an area. While too few nodes may not be as effective in a cooperative search of the area, too many will likely overwhelm the communication resources reducing the search efficiency. Communication within the swarm depends on the availability of neighboring nodes. The throughput in a network with a low UAV density will be low due to the lack of available neighbors. However, the throughput of a network with a high density may suffer due to the large demand for the shared medium.

Faster transmission of node information to neighbors decreases the effects of mode mobility, since a node can transmit more while a particular neighbor is in range. While longer communication ranges mean that more nodes can communicate with one another, shorter communication ranges can allow more than one node to effectively communicate if the sets of communicating nodes have sufficient distance between each other. This spatial separation allows more communication, which would have a beneficial effect on the efficiency of the search algorithm since more information is communicated in local areas. However, since UAVs are biased towards traveling in a single direction (discussed in Appendix A), the local sharing of information may not be as beneficial and could have an adverse effect on the search algorithm. Traveling in straight lines tends to cause UAVs to travel to relatively distant areas. A UAV arriving in new territory may be unaware of neighboring UAVs and which areas have been searched recently when shorter communication ranges are used. If information, such as the location of neighboring UAVs and the last time an area was searched, is not available, other UAVs may search areas previously searched, thus diminishing the efficiency of the group and increasing the time taken to search a given area.

3.1.2 Approach

At a high level, simulation is used to determine performance under different operating conditions. The effects associated with realistic wireless communications upon a group of UAVs using a distributed search algorithm are incorporated. However, instead of performing the simulation in MATLAB, as done in previous work [YPH06] and [PYT05], OPNET Modeler 10.5A is used. Measurements taken during simulation are analyzed to evaluate the impacts of the number of UAVs, available data rate, and effective communication range on the UAV swarm's ability to effectively scan a square search area.

3.2 System Boundaries

The system under test (SUT) is a cooperative UAV search system. The system includes a collection of UAVs, multiple ground targets, and a shared-channel data communications network contained within a specified search space. The distributed search algorithm is also part of the system and affects how each individual UAV moves within the search space. Figure 7 shows the cooperative UAV search system. The search algorithm is the component under test (CUT).

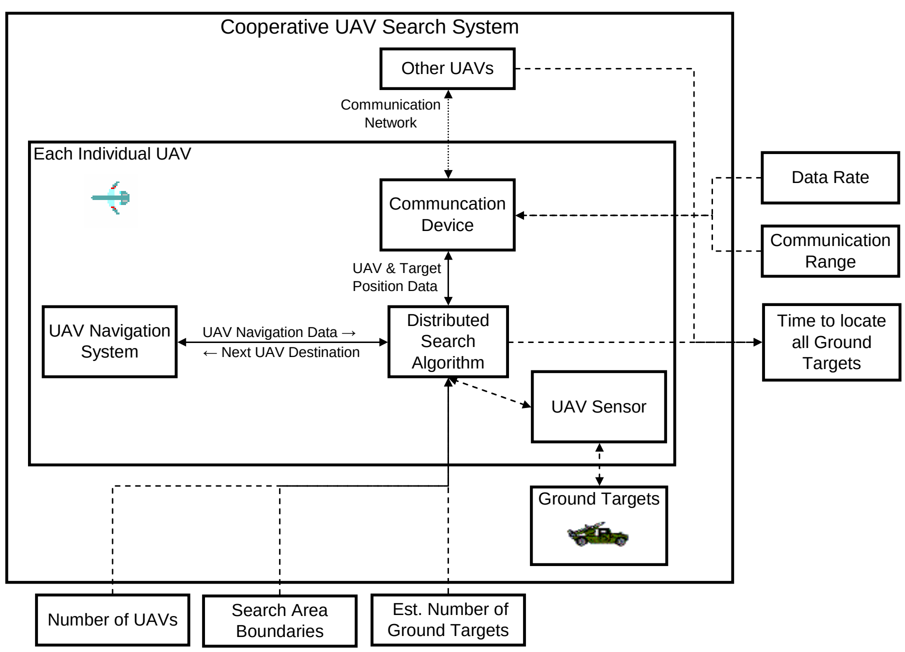

Figure 7. The Cooperative UAV Search System

Several aspects of this problem have been scoped to make the problem manageable. First, the UAVs and ground targets are restricted to a bounded two-dimensional plane. The UAVs' altitudes are fixed at 250 meters. The search area is free of obstacles or hazards, and UAVs may occupy the same physical two-dimensional location. No mid-air collision avoidance techniques such as altitude separation are considered. Each UAV has the ability to accurately determine its own position and antenna radiation patterns are isotropic. Finally, the number of ground targets and number of other UAVs in the group is known to each UAV at the beginning of the simulation; however, UAVs are unaware of the positions of the targets and are unable to locate their neighbors by a means other than communication.

3.3 System Services

The system provides a search service. This service is successful when all targets have been located within time constraints determined by the user. Upon success, the time taken to search the area is determined and is a measure of the system. Failure occurs when all targets are not located within the time constraints. Failures of the search service are not of interest and are also unlikely to occur as more time can be given to the system to complete the search task. Failures due to extremely long search times are assumed rare and constitute outliers.

The system also provides a data transfer service between UAVs. A successful outcome of the data transfer service occurs when the data reaches the intended destination uncorrupted. There are two different failure possibilities: (1) the data does not reach the intended destination or (2) the data is corrupted when it does reach the intended destination. The type of failure is not considered. Either failure is considered a failure of the data transfer service. The rate at which data can be successfully sent and received as well as the time it takes the data to successfully reach its destination are of primary interest.

3.4 Workload

The workload of the system is the information communicated between UAVs and used by the search algorithm. The number of UAVs, their speed, the area that can be sensed at a given time, the number of targets, and the boundaries of the search area are parameters that affect how much data is sent across the communication channel. The flow of information between nodes affects the decisions made by the search algorithm. A greater number of UAVs should produce more input for the search algorithm. With fewer UAVs, the system likely takes longer to complete the task of searching the area, and each UAV must work longer. More position data is produced as UAVs traverse the search area faster, which translates into a greater workload for the system.

The search algorithm divides the search area into smaller regions called cells. Once a UAV has scanned a particular cell it then communicates with the group. The area a UAV can sense drives how large cells are in the UAV's virtual grid since a UAV must be able to scan the entire cell when it is near the center of the cell. UAVs that can sense larger areas need fewer, larger subdivisions in the search area and thus communicate which areas have been searched less often than UAVs with smaller sensor areas. Like the number of UAVs, the size of the entire search area contributes to the length of the search. If the number of UAVs and their speed are constant and the search area size is increased, on average each UAV must search a greater area, which in turn takes more time. Lastly the number of targets also contributes to the workload. When targets are being tracked, the UAVs must share information about the targets to refine the UAVs' estimate of the target's location. The termination condition for each simulation occurs when all targets are located and eliminated. When a greater number of targets are present, more information about the targets is generated, and for a given number of UAVs, it is more likely that more targets will take longer to locate.

3.5 Performance Metrics

Two categories of performance metrics are measured: search performance and communication performance. When determining whether a particular configuration is better than another, the search performance metrics are the most important since the user is most concerned with how quickly and efficiently the area was searched. The performance of the network is also of interest because determining the scalability of the communication protocol is also one of the goals. If more UAVs can communicate effectively, then it is possible to add UAVs to cooperatively search an area of the same size without exhausting the available communication resources. The following metrics measure search performance:

- Search time: Search time is the time it takes from the beginning of the search until the UAVs locate all of the targets in the search area.
- Search redundancy: Search redundancy measures how many times a given cell is searched. In a completely efficient system where targets are immobile, all cells are scanned once. In less efficient systems, some areas are scanned more than

once. Because the total number of scans is important, the mean of the number of scans for all sections is relevant. The metric is the arithmetic mean of the number of times each area is searched.

The following metrics are used to evaluate the performance of the communication network:

- <u>Throughput</u>: Throughput is the measure of the number of packets transmitted successfully per unit time. Throughput is an indicator of how well a network accommodates the offered load. When data transmissions are low and the network is being infrequently used, throughput is naturally low. When the network becomes highly congested due to excessive demand, the throughput again can be low.
- <u>Number of Packets Sent</u>: The number of packets sent per unit time is a complementary measure to throughput. Since all packets are broadcast, multiple copies of the same packet can be received. Combining this information with throughput can give an indication of how many UAVs are engaged in communication activities on average.
- <u>Number of Collisions</u>: Network collisions occur when a receiver is in range of more than one transmitter that is sending data such that the multiple messages interfere with one another and become unintelligible to the receiver. Knowing how often collisions occur indicates the congestion of a network. A higher

number of collisions indicates that the network is operating less efficiently than a network with fewer collisions.

- End-to-End (ETE) Delay / Latency: The time delay between when data is sent from the source and received at the destination is known as ETE delay or one-way latency. ETE delay measures how long it takes for a node to send a message to another node. High latency, caused by high network congestion, diminishes a swarm's ability to communicate; potentially increasing the time it takes to search the area.

3.6 Parameters

Parameters (both system and workload) are the properties of the system that, if changed, are likely to affect the performance of the system.

3.6.1 System Parameters

- UAV mobility profile: The node mobility profile is the characterization of how nodes move throughout the search space. The nodes could follow a random mobility model; however, the search algorithm is used to determine the mobility profile in an attempt to minimize the search time. UAVs are assumed to be at random initial positions and have no initial knowledge of the status of other UAVs.
- Type of UAV: The type of UAV used affects many sub-parameters. Specifically, flight characteristics such as elevation and turn radius are determined by the shape and construction of the UAV. The type of UAV modeled in this study is the

Desert Hawk. Each UAV is configured with the same characteristics and parameters.

- Communication network: The communication network has several sub-parameters. The MAC protocol and routing protocol affects how information is disseminated through the network. The 802.11b protocol is used for the MAC and physical layer protocols. In this study, all traffic is broadcast and no particular routing scheme is implemented. Information to be forwarded, such as when a target has been located, is disseminated by flooding the network.
- Data rate: The data rate determines how fast data can be sent from node to node. Faster data rates can increase the throughput of the network, which results in increased cooperation and search efficiency of the swarm.
- Frame size: The frame size determines the transmission time between UAVs. The frame size is partially determined by how many position entries it contains and other data that must be included in the packet. A UAV may send several position entries in a single frame to diminish the overhead of sending the information across the network. Other frame components such as error detection and correction increase the frame size. The frame size for stage 1 and 2 packets is 96 bits, while stage 3 packets are 144 bits.
- Communication channel: The communication channel also has many subcomponents such as the frequency used, the attenuation, transmission power or range, antenna pattern, and bit error rate. Channel effects such as fading and multipath are subcomponents. All of these items determine which nodes can

communicate effectively enough to transfer data. This in turn affects overall communication and cooperation between nodes. By default, the 802.11b protocol in OPNET Modeler takes into account all of the items mentioned except fading and multipath effects.

- Communication Range: The communication range is determined by sub-parameters such as transmission power and receiver sensitivity. While a large communication range might cause more nodes to receive a single transmission, it may also limit the number of transmissions that can be made by other nodes within the range of the transmitter.

3.6.2 Workload Parameters

- Number of UAVs: The number of UAVs searching a space is normally determined externally based upon many different requirements. For instance these requirements could include the total inventory of UAVs and how many are allotted for searching several different areas. The total search time is likely to be less when more UAVs are searching the area.
- UAV speed: Search time is affected by the speed each UAV travels since more ground can be covered by a faster moving UAV. In terms of network traffic, more ground covered corresponds to an increase in network traffic. The speed of the UAVs in this study is 25 meters per second which is roughly 50 knots or 90 kilometers per hour.
- Search space: The size of the search space affects scan time. The shape of the search space may have some impact as well. For instance, nodes in a long,

skinny, rectangular region are more likely to be at the edge of the search space more often than UAVs in a square region of the same area. The direction the node travels next is affected by whether or not the node is near an edge of the search space. The search area used in the experiments is a square with 10 kilometer sides.

- Sensor area: The area the sensor can scan impacts the search process. The sensor area determines how much of the search space can be scanned by the UAV in a particular location. The sensor area and search area subdivisions are related such that the sensor can scan the entire subdivision when the UAV is near the center of the subdivision. The sensor radius is set at 65 meters, and the cells are squares with sides of 50 meters.
- Number of targets: Though realistically this parameter might not be known and the search could continue indefinitely, this value determines when an area has been cleared. The number of targets in each simulation is fixed at 25 and targets are not able to leave or enter the search area during the simulation. Once all targets are located the search is terminated.
- Target Characteristics: Target characteristics affecting performance include the mobility of a target and whether or not it can be sensed at a given instant. Mobile targets can move into previously-searched areas increasing the time it takes to find that particular target. Since targets can only to be sensed intermittently (e.g., when transmitting) a UAV may not detect a particular target.

3.7 Factors

Factors are a subset of the system and workload parameters. Factors are intentionally varied to determine the impact they have on system performance. The factors chosen are listed below and summarized in Table 2.

- Number of UAVs: The number of UAVs affects how fast an area can be searched as well as how much network traffic is generated. At most, 118 UAVs, sending one packet per second, should be able to effectively communicate at a data rate of 9.6 Kbps. Under realistic conditions fewer may be able to communicate [PaM03]. Based upon this analysis, the levels selected are 20 UAVs, 50 UAVs, 100 UAVs, 200 UAVs, and 300 UAVs.
- Data rate: The four data rate levels chosen are 1 Kbps, 10 Kbps, 100 Kbps, and 1 Mbps. These rates were chosen to reasonably fall within the range of 1.0 Kbps and 10 Mbps, which seem to be appropriate limits when surveying airborne communications. The 10 Kbps rate is a scenario similar to one previously studied [PaM03].
- Communication Range: The communication ranges chosen are 100, 1000, 3000, 5000, and 10000 meters. These ranges allow for a variety of communication conditions from essentially no communication between nodes (100m range) to allowing UAVs to communicate across nearly the entire search area (10 km range) in order to produce a scenario where nodes essentially have global knowledge.

Table 2. Chosen Factors

Factor	Level 1	Level 2	Level 3	Level 4	Level 5
Number of UAVs	20	50	100	200	300
Data Rate	1 Kbps	10 Kbps	100 Kbps	1 Mbps	
Communication Range	100 m	1,000 m	3,000 m	5,000 m	10,000 m

3.8 Evaluation Technique

The evaluation technique is simulation performed using OPNET Modeler 10.5.A. There are several reasons that simulation is chosen for the evaluation technique as opposed to analytic models or actual measurements. First, there are no general analytic models for sensor swarms. This is also true for the underlying MANET generalization of the sensor swarm from which to base an analytic model. Secondly, there are not many actual swarming or cooperative UAVs in existence, making actual measurements infeasible. At this time, most UAVs operate individually, are remotely piloted, and require modification to be autonomous and operate in a swarm environment. UAVs are also expensive and obtaining enough to perform this experiment would be difficult. It is not likely that results obtained though actual measurement could be easily reproduced based upon the current logistical issues of implementing a UAV swarm. Because simulation gives repeatable results in a controllable environment with reasonable cost and time constraints, simulation is the chosen evaluation technique.

The cooperative search rules are implemented in OPNET and verified by checking the calculations and decisions made during each stage of the process. This same verification method is applied to the ground targets and background processes. The correct operation of the models is also verified visually using the animation portion of the network simulator, which will allow for a general, large-scale verification.

The behavior of the communication network must be validated. This is done by comparing the network measurements between a scenario with UAVs where no targets are present and a scenario where generic network nodes, with traffic generation parameters that approximate the traffic generated by the UAVs, are used. These measurements are validated against those from similar scenarios studied for the IEEE 802.11b protocol. For instance, increasing the number of UAVs communicating with each other generally causes greater congestion on the network and correspondingly increases the end-to-end delay. The throughput of the network should also change. If the network was previously underutilized the throughput should increase. Otherwise, the throughput should decrease when the capacity of the network is exceeded.

3.9 Experimental Design

The experimental design is a full factorial design. Two of the factors have five levels and the other has four levels. This means a minimum of $5 \times 5 \times 4 = 100$ experiments are required. This does not include necessary replications to characterize the random effects. One measurement for each configuration is not enough to reasonably determine that one system is better than another; hence, more than one measurement must be taken to statistically compare the two systems. In each replication of an experimental configuration, the value used to seed the random number generator is changed. The value of the seed affects numerous items within the simulation including the initial placement of the UAVs, the random distributions necessary for the communications network, and the movements of the target nodes.

Confidence intervals are often used to compare the average behavior of systems. When the desired level of significance, the desired size of the confidence interval, and the coefficient of variation are known, the number of replications can be estimated. In the initial studies preformed it was determined that five replications of each experimental configuration were sufficient for the network metrics to be statistically different due to the small variance observed. However, five replications were not enough to observe statistical differences in the search performance. Ten replications provided a reasonable variance to observe statistical differences in the data and allowed all of the simulations necessary to be run in a reasonable amount of time. Because the search performance metrics are of greatest interest, the number of replications is decided by these metrics. The total number of simulation runs required is 100 experiments × 10 replications = 1000. The seed values that will be used are (in order of use): 899, 429, 200, 303, 538, 910, 525, 309, 34, and 715. In case a failure of the search service occurs, the seed values used to perform extra replications are: 769, 60, 627, 265, and 312.

3.10 Analysis and Interpretation of Results

Because random processes are used within the simulation, the data collected are random variables. The accuracy and precision of the measurements must be taken into account for the purpose of comparing systems. Multiple measurements with the same configuration will likely result in a truer representation of how the system actually performs. Errors in the sampled data are assumed to be the result of a normal distribution and confidence intervals are constructed to compare systems. Systems with non-overlapping confidence intervals can be said to have a significant statistical difference. However, with overlapping confidence intervals, where the mean of one system is within the bounds of the other confidence interval, the only thing that can be said is that the two systems are not statistically different. This does not mean that the systems perform the same, but rather based upon the measurements taken, neither system can be said to perform better or worse than the other. When the two intervals overlap and the mean of either system's performance does not lie within the bounds of the other interval a t-test can determine if the systems are significantly statistically different or not.

A more powerful method of analyzing the results is analysis of variance (ANOVA). The strategy behind ANOVA is to quantify and separate the variation in the data that is due to the randomness associated with measuring the data and the variation that is due to actually changing the factors in the experiment. ANOVA is used to analyze the data to determine which factors most affect performance. However, to use an ANOVA, assumptions about the data must be verified so that the results have meaning. These assumptions include those mentioned above as well as some that are implied. One of these assumptions is that errors in the data are normally distributed. Another assumption made is that the standard deviation of the distribution remains constant. These assumptions can be verified by inspection of the data and plots of both the data and the residuals due to the error.

3.11 Summary

This chapter discusses the methodology to determine the impact of increasing the number of UAVs communicating at various data rates and effective node communication ranges. The experimental technique is simulation using OPNET Modeler measuring search time, search redundancy, network throughput, packets sent, the number of collisions, and network latency. The factors varied include the number of UAVs, the network transmission rate, and the effective communication range of the UAVs.

IV. Analysis and Results

This chapter presents the results and analysis on the data collected from the simulation scenarios. Section 4.1 presents the validation of the modified IEEE 802.11b network performance with no targets present. Section 4.2 presents the results collected for each of the search performance metrics and the network performance metrics, discussed in Chapter III, as well as provides analysis on the results. Section 4.3 analyzes the data as a whole and provides some real-world analysis of the data.

4.1 Network Validation

The purpose of this section is to validate the communication network by comparing the results obtained from generic wireless nodes provided by OPNET and the UAV models developed for this study. Part of the validation of the UAV models assumes that the communication network results produced by a commercial network simulation tool are reasonably accurate. However, the results obtained are also validated against those reported in other studies.

The plots presented and discussed in this section as well as the additional plots provided in Appendix B have a common format. Because the intent of these plots is to compare the trends of two data sets, two different line styles are used. Solid lines show the data collected from the generic OPNET models, while dashed lines represent the data collected from the UAV models. Solid and dashed lines with the same color use corresponding factors and levels. The figures are composed of four subfigures, which are used to distinguish different data rates.

In general, data from the two models exhibit the same overall trends. Confidence intervals were calculated for the means that were plotted; however, most of the samples have very small variation from the sample mean and thus the confidence intervals are very narrow. When the confidence intervals are plotted, they are hard to distinguish because the small size of the interval. To avoid confusion, the confidence intervals are not plotted. Most of the corresponding points between the two data sets are significantly statistically different at the 0.1 level of significance; however, it is the fact that the overall trends are similar that is most important.

When targets are not present, the rate at which packets are sent by each UAV depends on the time it takes the UAVs to scan each subdivision of the search area. When a UAV completes scanning a cell, the UAV chooses its next destination and sends a single packet to communicate its next destination to neighboring UAVs. The total rate packets are sent by the network is the product of the packet rate per UAV and the number of UAVs sending packets. Because the search area subdivisions are rectangular and the UAVs move at a constant velocity, the mathematical interval of packet rates per UAV, P, is approximated by:

$$P = \left[\frac{v_{UAV}}{\sqrt{c_x^2 + c_y^2}}, \frac{v_{UAV}}{\min(c_x, c_y)} \right] \tag{4.1}$$

where v_{UAV} is the speed of the UAVs, c_x is the size of x-dimension of the individual cells in the search space, and c_y is the y-dimension of the individual cells. The reason for the interval is that it takes longer for the UAVs to move to a diagonally adjacent cell than a regularly adjacent cell. If the UAVs were choosing destinations at random from the eight surrounding cells (i.e., without the search rules), it is equally likely a UAV would choose to visit a diagonal or adjacent cell next. To determine the average rate packets are sent, the velocity of the UAV is divided by the average distance between cells, both diagonal and adjacent, which is the average of the two denominator expressions. Therefore, the average rate packets are sent without targets present is modeled using the following equation:

$$\overline{P} = \frac{2 \cdot v_{UAV}}{\min(c_x, c_y) + \sqrt{c_x^2 + c_y^2}} \tag{4.2}$$

In the network of generic nodes, packets are uniformly generated per UAV at a rate that matches the interval P defined in Equation (4.1). The average rate packets are sent is presented in Figure 8. The black solid line is the product of the average packet rate, $\overline{P}$, and the number of UAVs, which yields the average rate that packets were generated for all UAVs. The dotted black lines are the product of the interval, P, and the number of UAVs; thus, giving a representation of the expected maximum and minimum values of the analytic rate at which packets are sent. The solid lines for the generic nodes are plotted in Figure 8; however, they match the black analytic line and cannot be seen. The dotted lines representing the UAVs deviate from the analytical average, but stay within the analytic region bounded by the black dotted lines. Figure 40, in Appendix B, plots the same data versus communication range. It can be seen in that plot, that the traffic sent by the UAVs also follows the same trends as that of the generic nodes. Though not plotted in Figure 40, the solid generic node lines match the analytic average packet rate. If the analytic lines were plotted, the generic node lines would not be visible.

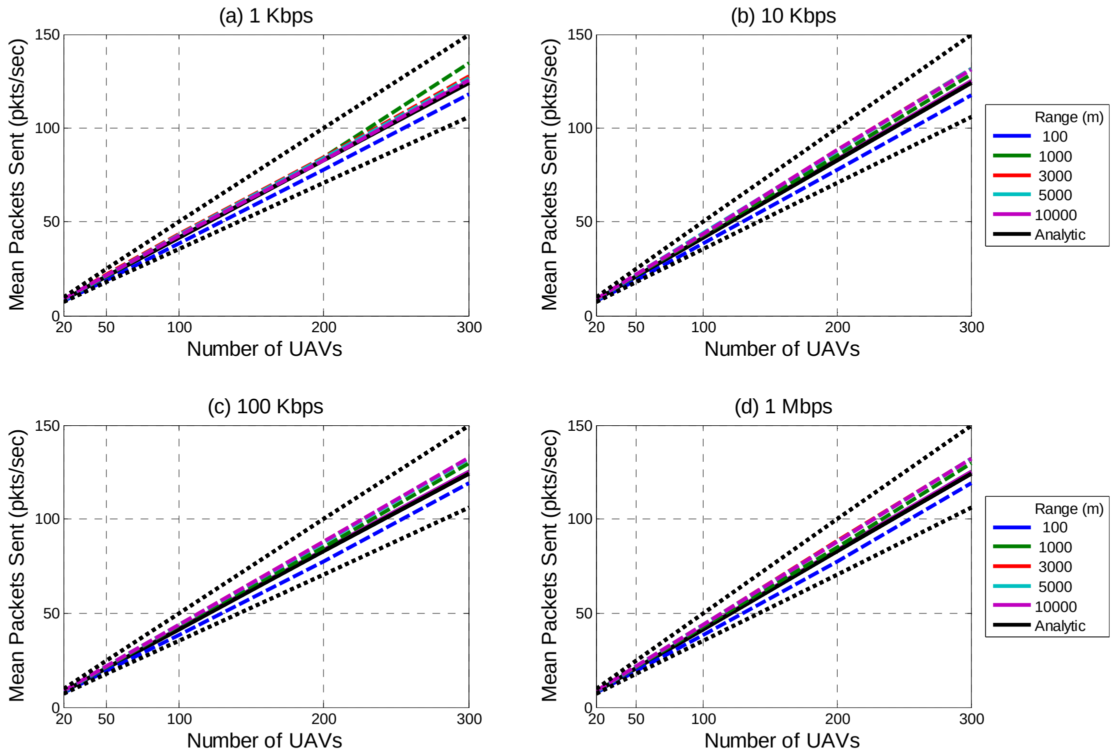

Figure 8. Mean Number of Packets Sent without Targets versus Communication Range

Figure 41 and Figure 42 show the trends in the mean throughput of the network reasonably match. However, in most studies, the throughput of the network is normalized by the amount of total traffic that can be sent across the network and plotted against the normalized load of the network. This is usually done at the individual node level. When done at this level the individual node throughput and load can be easily normalized because only one packet can be sent or received correctly at a time. The data collected is the average throughput for the entire network and must be divided by the number of nodes to get an average throughput per node. However, there is a problem with this when simultaneous communication occurs in the network due to spatial multiplexing. Since this situation can arise when communication ranges less than the full range (10 km) are used, only the 10 km range can be normalized. A plot of normalized throughput versus normalized load is presented in Figure 9.

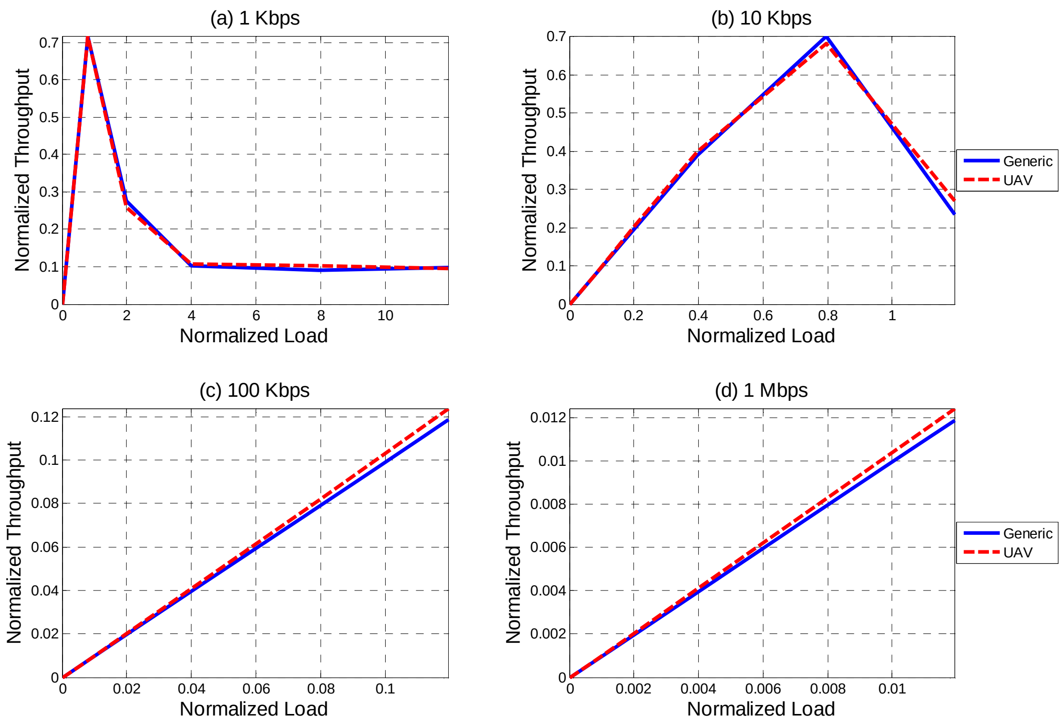

Figure 9. Normalized Throughput without Targets versus Normalized Network Load at the 10,000 meter Communication Range

A survey of several studies, [Bia00], [CVB02], [HoC96], [LHP05], [Swe99], and [TaK85], suggests that the maximum normalized throughput of a network similar to this configuration should be between 0.6 and 0.8. In the cases where the throughput is plotted against normalized load, the throughput rises until it peaks near or slightly before 1.0 and diminishes quickly thereafter. Most of the studies cited use data rates greater than 1 Mbps and packets of much larger size. No one study matched the specifics of this study perfectly. Figure 9(a) and (b) match well and these peaks occur close to an offered load of one.

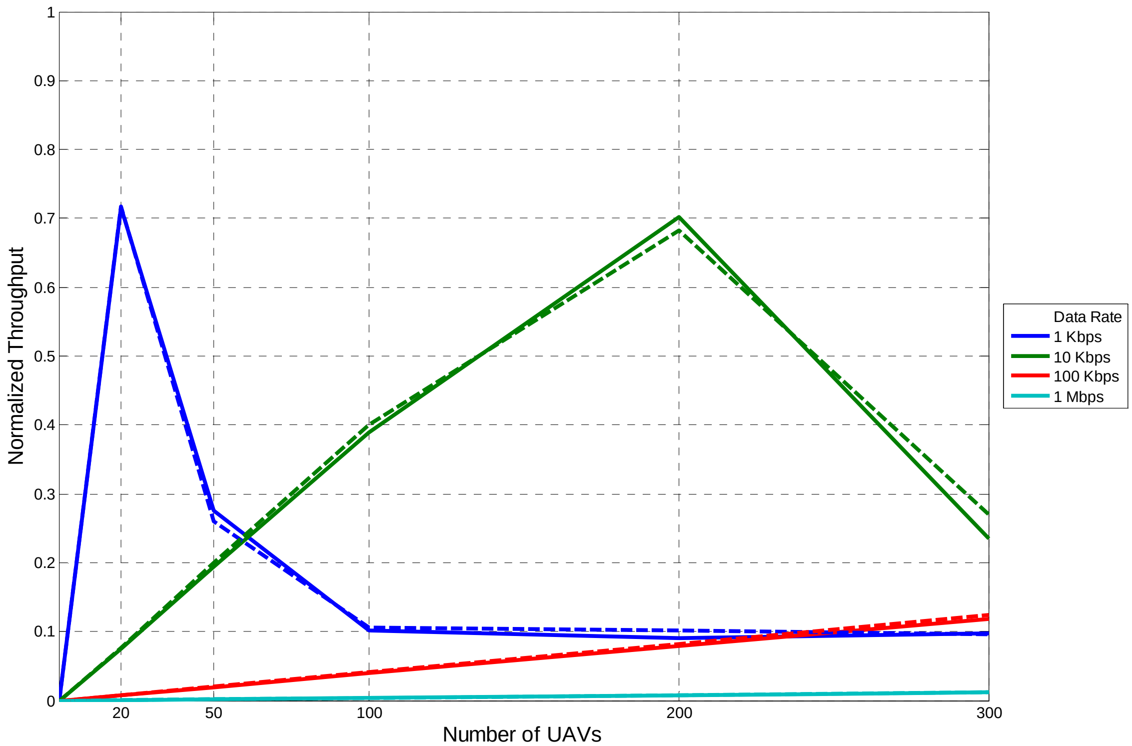

Figure 10. Normalized Throughput without Targets versus Number of UAVs at the 10,000 meter Communication Range

Pack and Mullins [PaM03] present an equation for the type of broadcast communication network used in these experiments. A modified form of this equation uses the packet rate rather than the node's speed, which are related in Equation (4.1), and substitutes the value of one for the position entries per packet. The resultant equation is given by:

$$n = \frac{R/\overline{P}}{S_P + (R \cdot IFS)} \tag{4.3}$$

where n is the maximum number of UAVs that can ideally communicate without collisions, R is the data rate of communication network, $\overline{P}$ is the average packet rate, S_P is the size of the packets in bits, and IFS is the distributed interframe space (DIFS) time used by the communications network. Using Equation (4.3), the values for each data rate are calculated and listed in Table 3. The values of *n* for the 1 and 10 Kbps data rates roughly correspond to the points of maximum normalized throughput in Figure 10. Since the value of *n* is an ideal value, it should overestimate the true number of nodes that can communicate making the peaks in throughput seem reasonable.

Table 3. Estimated Number of Nodes (*n*)
That Can Communicate at Different Data Rates

Data Rate	*n*
1 Kbps	25
10 Kbps	248
100 Kbps	2,219
1 Mbps	10,778

4.2 Results and Analysis of Individual Measures

This section presents and analyzes the data collected from the experiments. Because the search performance is of greater importance than the network performance, the search performance metrics are discussed separately, while the network results are all discussed in one section. Section 4.2.1 focuses on the search time performance of the UAVs, while Section 4.2.2 focuses on the search redundancy metric. Section 4.2.3 discusses the network performance.

4.2.1 Analysis of Search Time

The mean search time is plotted against communication range in Figure 11. The confidence intervals for a 0.1 level of significance have been included in the graph. From the inspection of these plots, it appears that few significant statistical differences exist between the communication ranges. All graphs in Figure 11 are plotted on the same scale so that it can be seen that the data rate also appears to have few significant statistical differences. However, there does appear to be many statistically significant differences between different numbers of UAVs. Though this is difficult to visually determine for the larger numbers of UAVs in Figure 11, it can be determined when examining the numerical values.

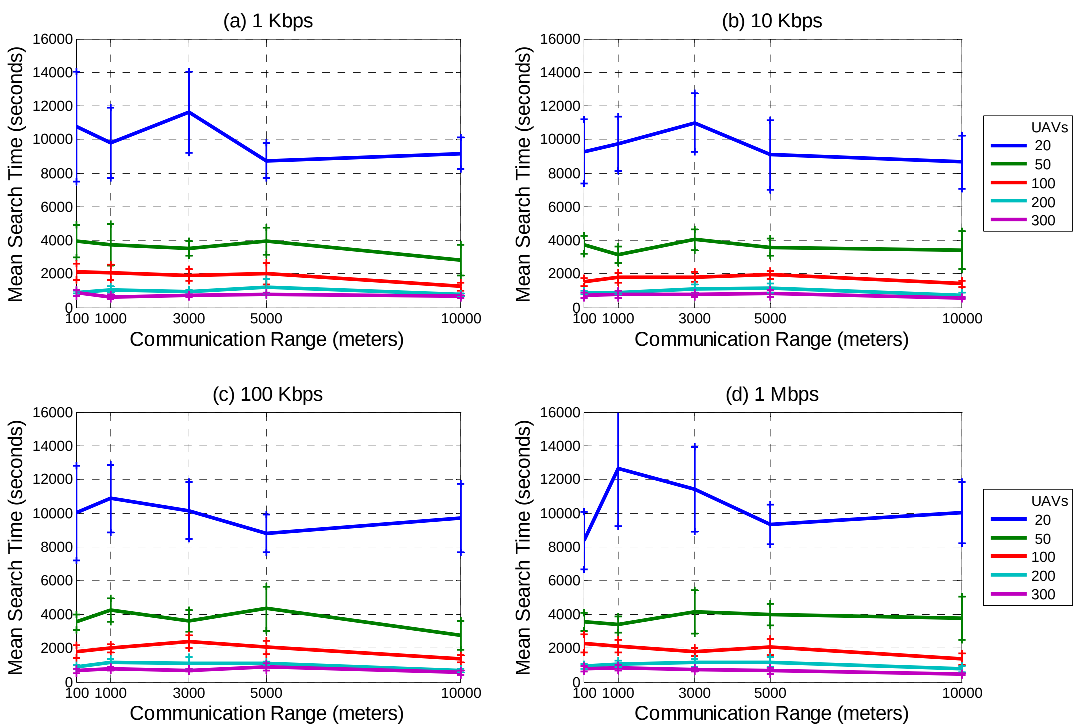

Figure 11. Mean Search Time versus Communication Range

Figure 12 presents the same data that is plotted in Figure 11. The difference between the two graphs is that search time is plotted against the number of UAVs, so that the relationship between the search time and number of UAVs can be more readily seen. The relationship appears to be an exponential relationship. From Figure 11 and Figure

12, it would seem that number of UAVs is the only significant factor in the time it takes the UAVs to find all of the targets in the search area.

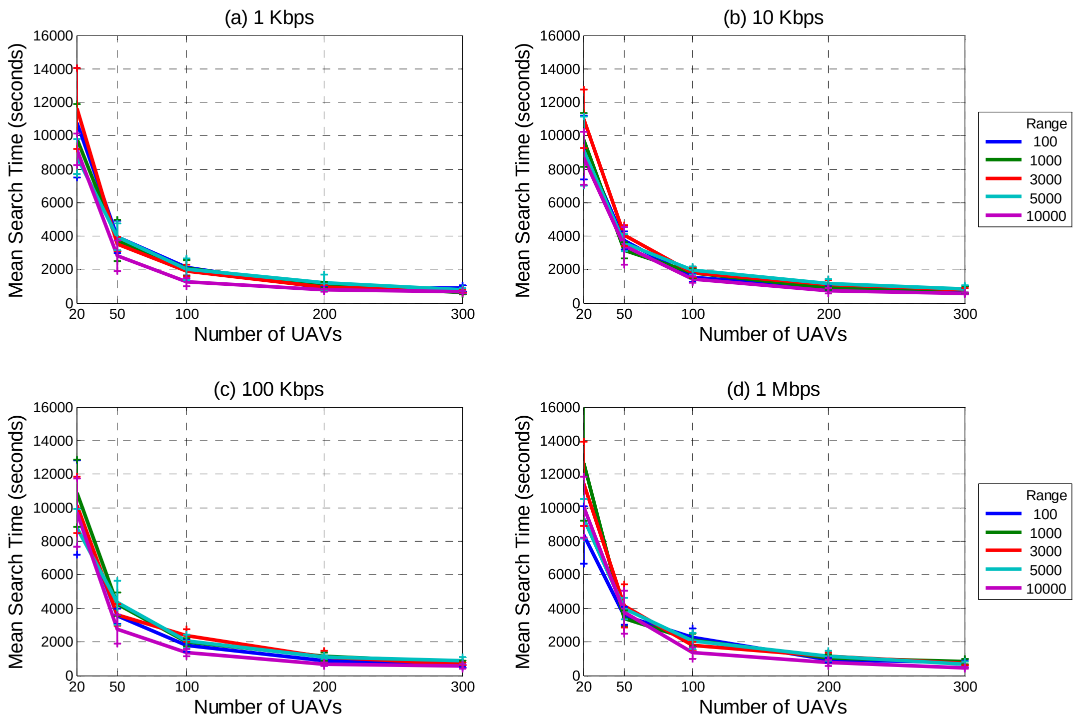

Figure 12. Mean Search Time versus Number of UAVs

To use an ANOVA the following assumptions must be verified: (1) The relationship of the data is linear, (2) errors are independent, (3) errors are normally distributed, and (4) the standard deviation of the errors is constant. When attempting to use an analysis of variance (ANOVA) on the search time results, it was found that the third and fourth assumptions for the ANOVA were not met, while the first assumption is questionable. However, a transformation on the response can yield data that does meet the criteria for a valid ANOVA. Figure 13 presents the same data as plotted in Figure 12, but plots the data using a logarithmic scale on both axes. Because the data form straight lines in this plot, it is confirmed that the relationship between number of UAVs and search time is a power function where the log of the search time is related to the log of the number of UAVs.

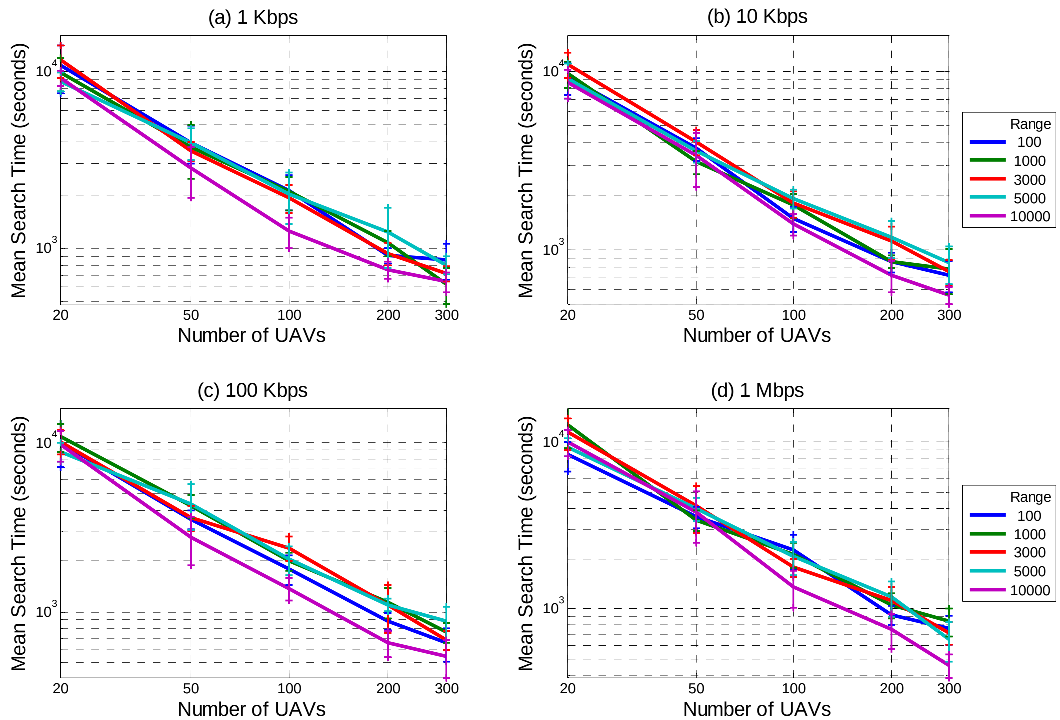

Figure 13. Mean Search Time versus Number of UAVs with a Log-Log Scale

When the log of the search time is analyzed using an ANOVA, all of the assumptions of the ANOVA hold. The results of the ANOVA are presented in Table 4 and visual tests to confirm the assumptions of the ANOVA are shown in Figure 14. In Figure 14(a) and (b) the blue line provides a reference as to what a normal distribution would look like. The fact that the residuals (errors) closely follow these trends verifies the assumption that the residuals are normally distributed. Figure 14(c) plots the residuals against the fitted values in the data. Because no obvious trend exists, the assumption that errors are independent is met. Also, because the deviation of the errors remains fairly constant and does not appear to increase, the assumption that the standard deviation is constant is verified. Lastly, from Figure 13, it can be confirmed that the data appears to follow a linear trend. Similar plots to those in Figure 14 will not be presented in future sections. However, whether the data conforms to the criteria for an ANOVA is noted.

In Table 4 it can be seen that 87.32% of variation in the results is due to changing the number of UAVs. From the ANOVA it can also be concluded that the number of UAVs, the communication range, and their interaction are statistically significant factors. Communication range alone accounts for 1.22% of variation while error accounts for 10.42% of the variation. Data rate and any interactions it has with other factors is not significant with respect to error.

Table 4. Results of Using an ANOVA on Log Search Time

Source of Variation	DF	Adj SS	% Variation	Adj MS	F Ratio	Prob > F
Number of UAVs	4	169.53	87.32	42.3816	1885.39	0.000
Communication Range	4	2.3721	1.22	0.593	26.38	0.000
Data Rate	3	0.0313	0.02	0.0104	0.46	0.708
Number of UAVs*Communication Range	16	0.7421	0.38	0.0464	2.06	0.008
Number of UAVs*Data Rate	12	0.1371	0.07	0.0114	0.51	0.910
Communication Range*Data Rate	12	0.3118	0.16	0.026	1.16	0.311
Number of UAVs* Communication Range*Data Rate	48	0.7929	0.41	0.0165	0.73	0.910
Error	900	20.231	10.42	0.0225		
Total	999	194.15				

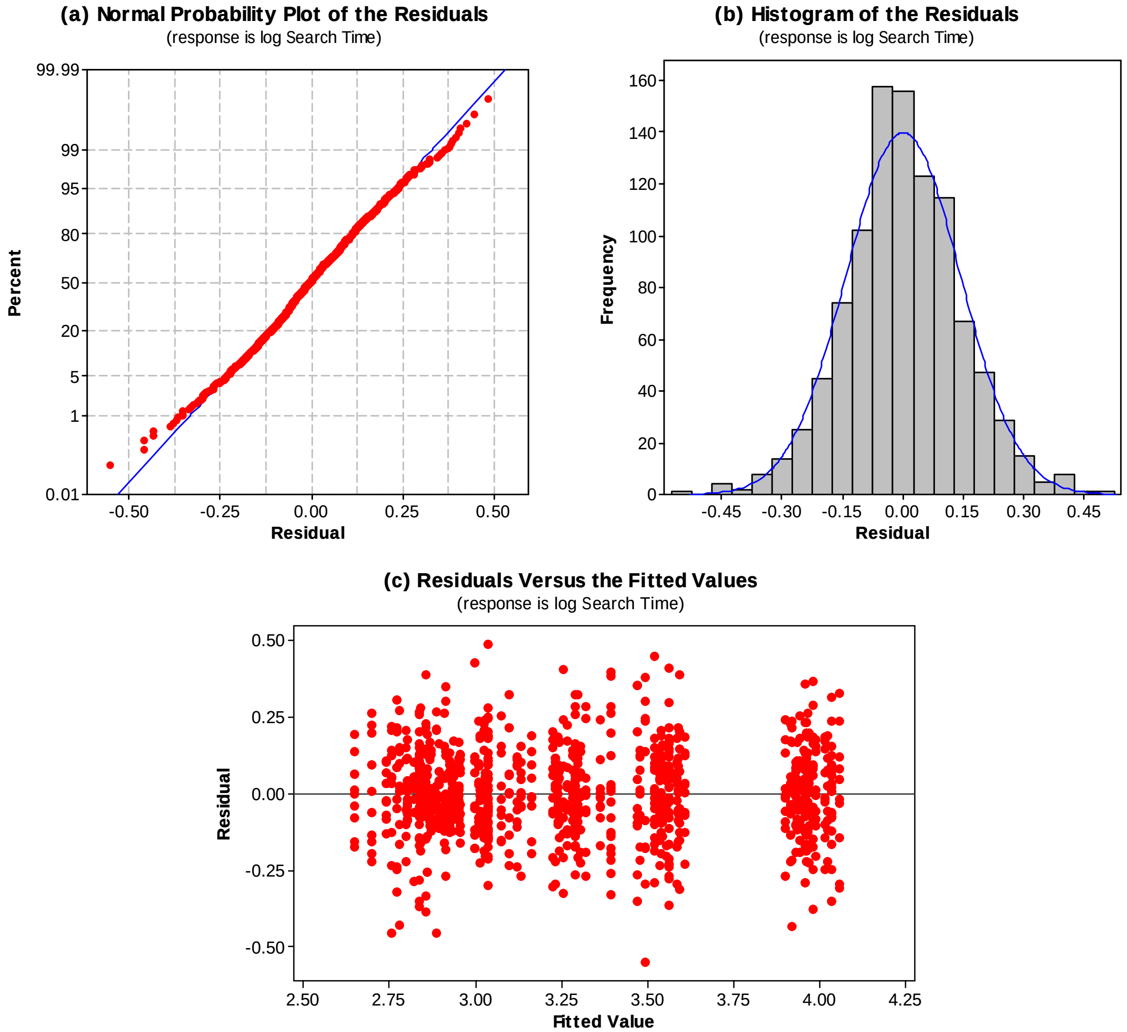

Figure 14. Plots for Verifying the Assumptions of the Log Search Time ANOVA

The effects of the significant factors and their interaction are presented in Figure 15. From these plots, the trend relating search time and number of UAVs is again confirmed. The effect that communication range and its interaction with number of UAVs appear to have on search time is that the largest communication range results in lower search times. Using pair-wise comparisons of the mean responses at the 0.1 level of significance, each level of number of UAVs has significant statistical differences from all other levels. The 10,000 meter communication range shows significant statistical differences from all other levels, but the other levels are not significantly statistically different from each other. Data rate comparisons yielded no significant statistical differences between factor levels.

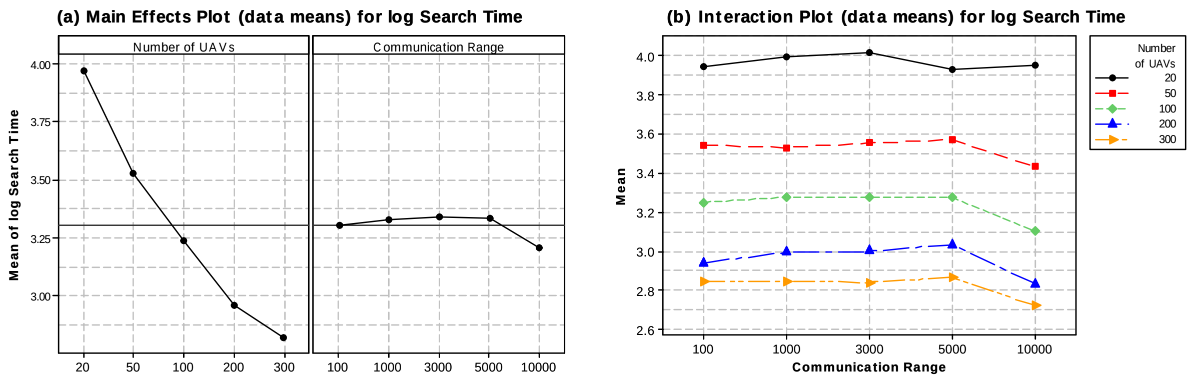

Figure 15. Log Search Time Main Effects and Interaction Plots for Significant Factors

A prediction of the number of UAVs necessary to search a space in a given time would be a useful model. Since most of the variation in the data was due to the number of UAVs, and since a relationship between the two appears to be exponential, a linear regression model to mathematically relate them is reasonable. The best fit line for the number of UAVs is given by:

$$\log\left(N_{UAVs}\right) = 4.905 - 0.892 \cdot \log\left(T_S\right) \tag{4.4}$$

where N_{UAVs} is the number of UAVs and T_S is the search time. The regression line has an R^2 value of 87%, which indicates it is a moderately good fit to the data and is useful for prediction purposes.

4.2.2 Analysis of Search Redundancy

Figure 16 is a plot of the mean search redundancy versus communication range. Confidence intervals at the 0.1 level of significance are also included in the graphs. From the graphs, it appears that the general trend for search redundancy is, except for the 20 UAVs case, the search redundancy rises and peaks at the 3,000 or 5,000 meter communication range and then decreases at the 10,000 meter communication range.

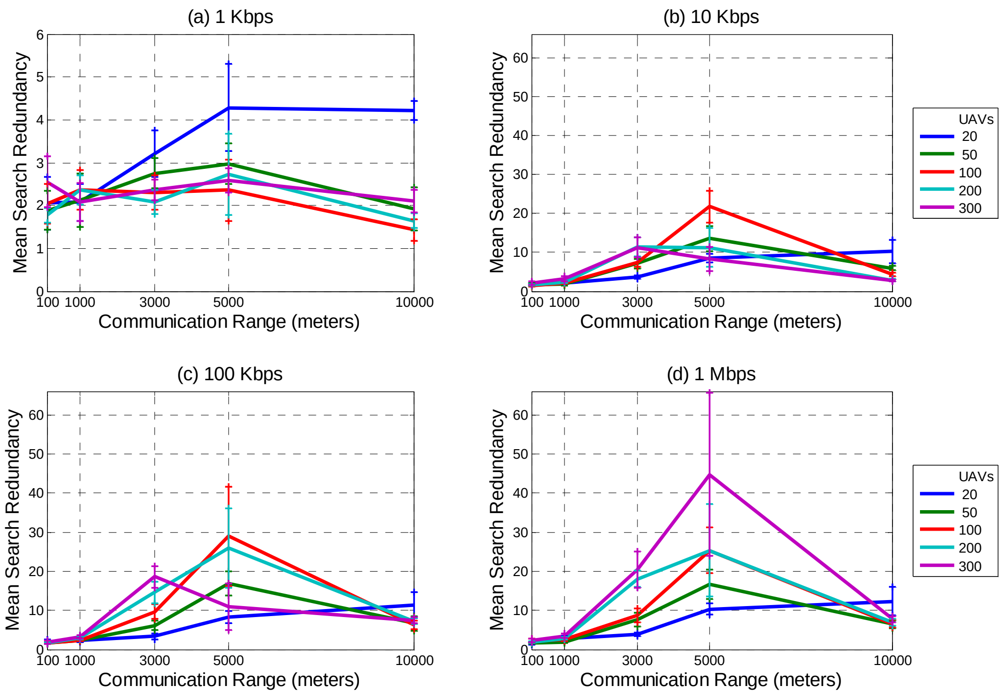

Figure 16. Search Redundancy versus Communication Range

Figure 17 is similar to Figure 16, however the data is plotted against the number of UAVs. When viewing the data from this perspective it can again be seen that the search redundancy tends to be greater for the 3,000 and 5,000 meter transmission ranges.

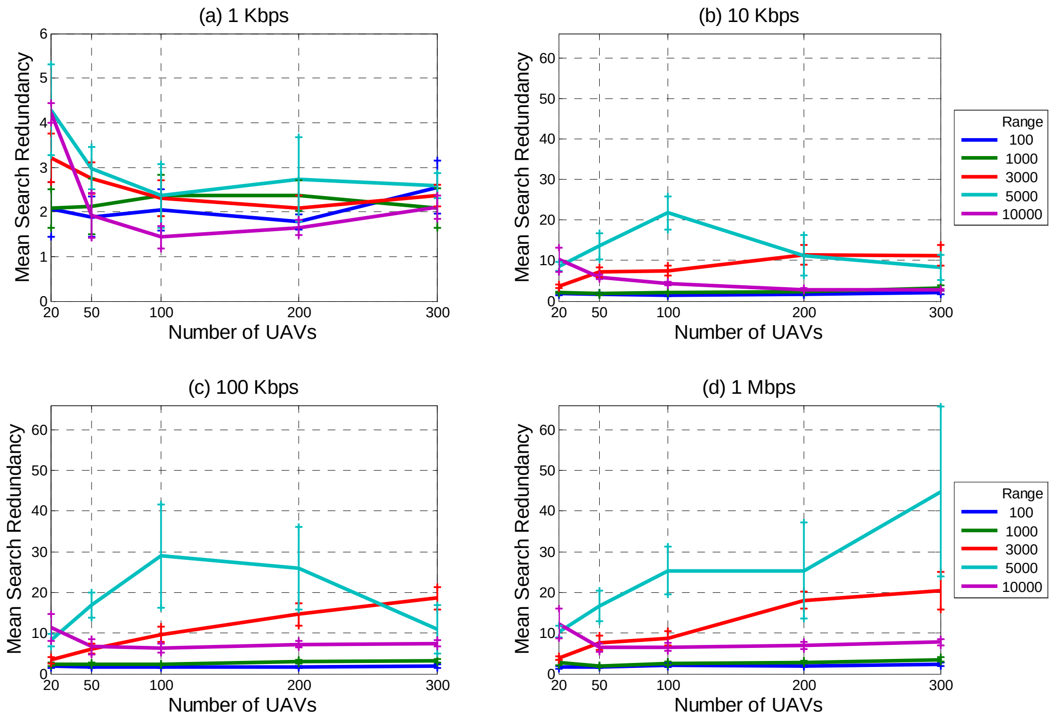

Figure 17. Search Redundancy versus Number of UAVs

The search redundancy data does not meet the necessary assumptions to perform an ANOVA on the data. More specifically, the residuals are not normally distributed. A log transformation produced data that met the assumptions for an ANOVA. Table 5 presents the results of the ANOVA on the log search redundancy. From the ANOVA it can be concluded that changes in the response due to all of the factors and their interactions are significant with respect to the error. The largest contributors of variation are the communication range (16.3%), data rate (45.01%), and the interaction of communication range and data rate (10.67%).

Table 5. Results of Using an ANOVA on Log Search Redundancy

Source of Variation	DF	Adj SS	% Variation	Adj MS	F Ratio	Prob > F
Number of UAVs	4	0.9743	0.63	0.2436	9.22	0.00
Communication Range	3	25.112	16.30	8.3707	317.01	0.00
Data Rate	4	69.364	45.01	17.3409	656.73	0.00
Number of UAVs* Communication Range	12	3.5029	2.27	0.2919	11.06	0.00
Number of UAVs*Data Rate	16	9.2628	6.01	0.5789	21.92	0.00
Communication Range*Data Rate	12	16.446	10.67	1.3705	51.9	0.00
Number of UAVs* Communication Range*Data Rate	48	5.6792	3.69	0.1183	4.48	0.00
Error	900	23.764	15.42	0.0264		
Total	999	154.11				

Figure 18 is a plot of the effects of each of the main factors as well as the interaction between data rate and communication range. A pair-wise comparison of mean responses indicated that significant statistical differences exist between the 20 UAV level and all other levels of UAVs except 50 UAVs as well as between the 50 UAV and 200 and 300 UAV levels. For communication range, significant statistical differences exist between all factor levels. All data rate factor levels are significantly statistically different from each other as well.

The trends observed in the search redundancy data are explained by the behavior of the UAVs once a target is detected. When UAVs orbit a target, they continue to scan the individual subdivisions of the search area. Because the UAVs move faster than the ground targets, it is likely that the UAVs in orbit will scan cells around a target multiple times in fairly rapid succession. At the 100 and 1,000 meter communication ranges it is unlikely that other UAVs will receive requests to refine a target's location and thus fewer UAVs enter an orbit around a target. This in turn lessens the search redundancy since other UAVs continue to globally search the scan area. At the 3,000 and 5,000 meter

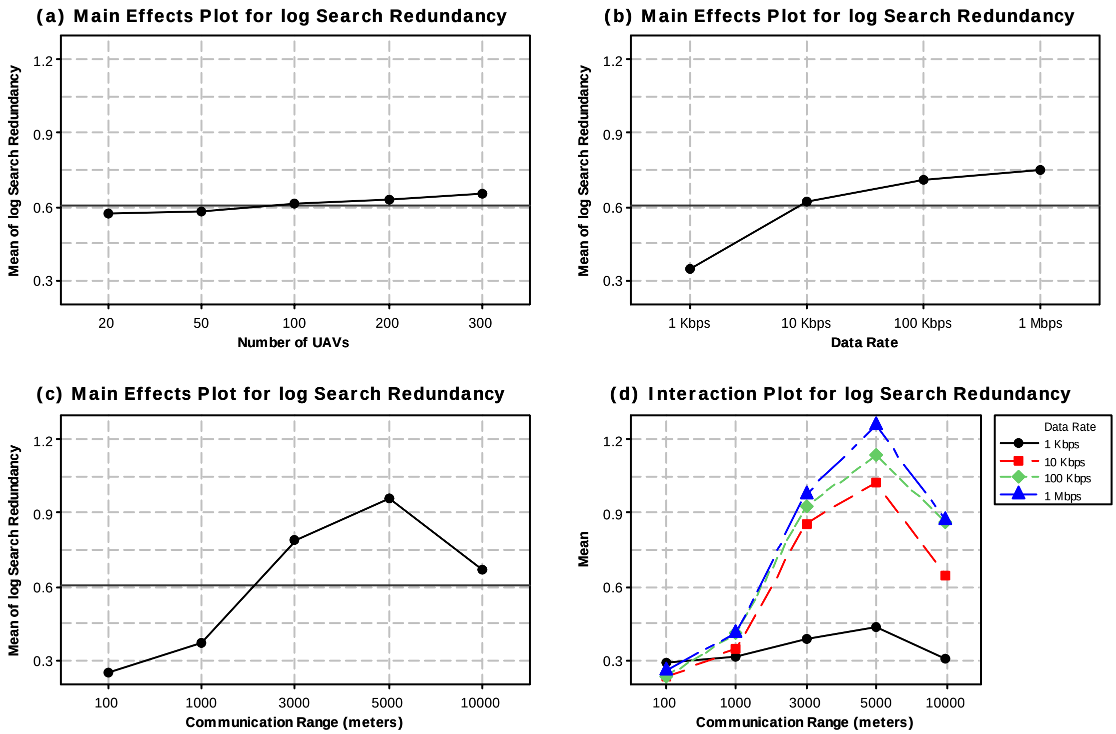

Figure 18. Main Effects and Selected Interaction Plots for Search Redundancy

communication ranges, other UAVs are more likely to respond to requests to refine a target's location and will enter orbit around a target. With more UAVs orbiting a target the search redundancy of these areas is likely to increase rapidly. It is likely that better coordination occurs at the 10,000 meter communication range and a more suitable number of UAVs attempt to orbit a target. However, another explanation for the slower data rates could be that the 10,000 meter range is not as well suited for communication as the 3,000 or 5,000 meter ranges. Because of this, the overall sharing of information between UAVs might be similar in performance to the shortest communication ranges causing fewer UAVs to attempt to assist in refining a target's location.

4.2.3 Analysis of Network Performance Measures

The following sections provide an analysis of the network performance. Each metric is discussed in its own section.

4.2.3.1 Mean Packets Sent

Figure 19 shows the rate at which packets are sent plotted against the number of UAVs. The confidence intervals shown are at the 0.1 level of significance. The trend is similar to the trend seen during the network validation presented in Section 4.1. The analytic range defined in Equation (4.1) is designated as before with two dotted black lines. Since the UAVs generate network traffic at a slightly quicker rate when orbiting targets, it is not surprising that the trends seen in Figure 19 are larger than the analytical range.

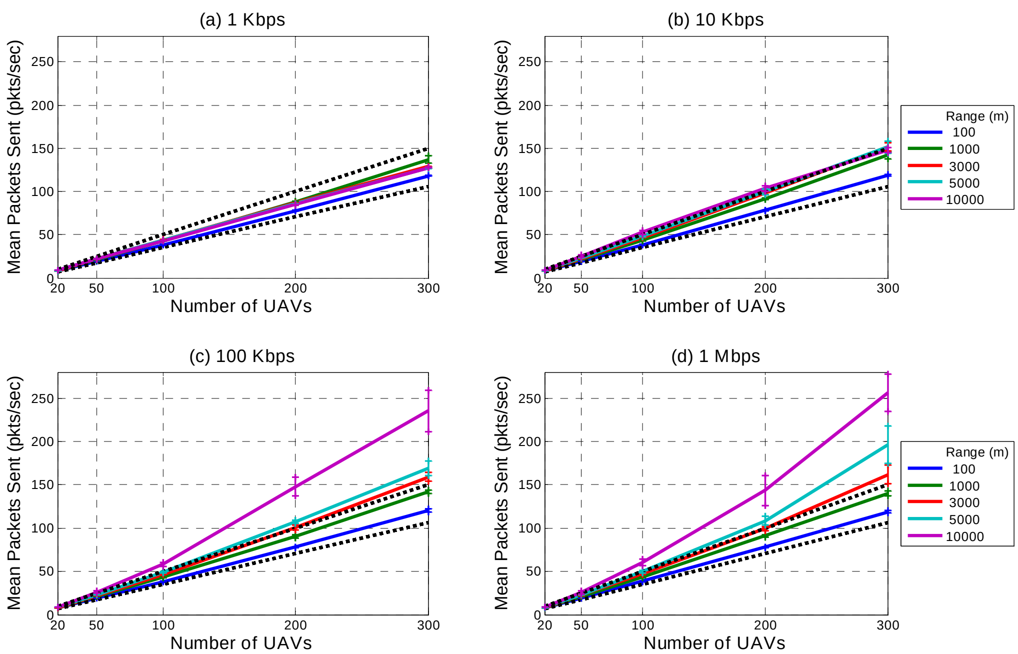

Figure 19. Mean Packets Sent versus Number of UAVs

Figure 20 presents a different view of the data shown in Figure 19. In Figure 20, the number of packets sent is influenced by the communication range and data rate. This trend is not surprising because, due to the presence of targets, when UAVs assist in refining a target's location, packets are generated at a higher rate. When a UAV enters orbit around a target, the UAV shares information about the target at a pre-defined rate of 0.5 packets per second in addition to sending information about the cells the UAV happens to scan while orbiting the target. When targets are not present, the average rate of communication is approximately 0.4142 packets per second. Furthermore, when the communication range is greater, it is more likely that other UAVs will be in range and receive a request to assist refining a target's location. Once in orbit, the additional UAVs will also begin transmitting packets at the higher packet rate.

When a target is located and eliminated, the network is flooded to notify the UAVs that the target was eliminated. The network must be flooded as no other routing scheme is in place and the number of targets remaining is important information that affects the behavior of the UAVs when evaluating whether or not to assist other UAVs in refining target locations. Each UAV that receives a flood packet rebroadcasts the packet once. This causes several copies of the packet to be generated in a relatively short amount of time and momentarily causes a large increase in the number of packets sent.

When communication range is greater, it is more likely that the flood will propagate across the entire network, either in a single hop or through a series of hops from UAV to UAV, and more copies of the original packet will be generated. For instance, at the 100 meter communication range, it is likely that very few UAVs will receive and forward the flood packet. Hence, the flood may end before reaching all nodes in the network. At the 10,000 meter communication range, it is likely that all nodes will receive the flood packet in a single hop and each node will attempt to send a copy of the information. At an intermediate range such as the 3,000 meter communication range, it is likely that several hops must be made before a distant UAV will receive the target elimination information. The number of UAVs comes into play in this situation. When more UAVs are present it is more likely that a UAV will be in range and will be able to successfully forward the flood packet to other UAVs.

Data rate also impacts both of the previous scenarios. When the network has a greater data rate, there is less congestion in the network and it is more likely that requests and floods are correctly received (i.e., a collision with another packet is less likely).

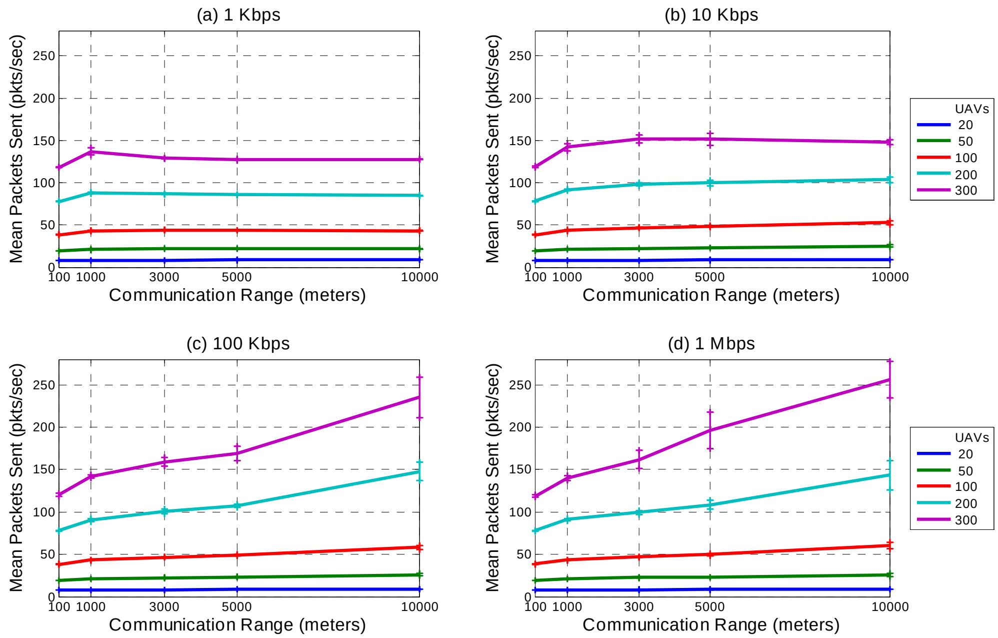

Figure 20. Mean Packets Sent versus Communication Range

The data does not meet the criteria for an ANOVA. Specifically, the residuals are not normally distributed and the deviation from the mean is not constant. Table 6, instead, provides the percent of variation due to the factors, interactions, and error. The number of UAVs accounts for most (87.32%) of the variation, while the other entries in the table, including error, each account for approximately 1% to 2% of the remaining total variation.

Table 6. Percent Variation for Packets Sent

Source	DF	Adj SS	% Variation
Number of UAVs	4	2822653	87.32
Data Rate	3	33395	1.033
Communication Range	4	81049	2.507
Number of UAVs*Data Rate	12	49111	1.519
Number of UAVs*Communication Range	16	77569	2.400
Data Rate*Communication Range	12	43402	1.343
Number of UAVs*Data Rate* Communication Range	48	61933	1.916
Error	900	63581	1.967
Total	999	3232693	

4.2.3.2 Mean Network Throughput

Figure 21 is the normalized throughput for the 10,000 meter communication range. The network throughput peaks at roughly the same values for the given number of UAVs as the validation examples. This plot suggests a network using a 1 Kbps data rate becomes more and more congested for more than 20 UAVs. Likewise, the same is indicated for the 10 Kbps data rate with more than 200 UAVs.

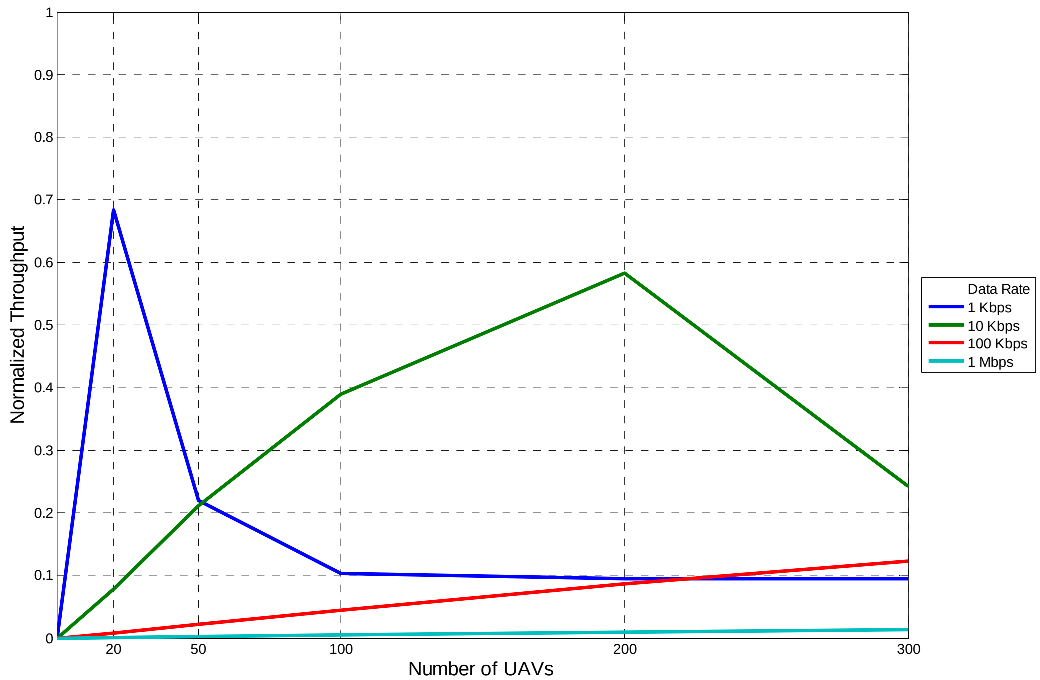

Figure 21. Normalized Throughput versus Number of UAVs
at the 10,000 meter Communication Range

Figure 21 presents the 10,000 meter communication range. The other communication ranges are presented in Figure 22. Figure 22 does not show normalized throughput because spatial reuse makes it difficult to properly normalize the throughput for communication ranges less than 10,000 meters. Figure 22 does include confidence intervals for the throughput at the 0.1 level of significance. The trends in the figure do not indicate any single factor dominates more than another. The narrow confidence intervals suggest that most of the changes in response are statistically significant. Figure 47 in Appendix C displays the same data as Figure 22 except that the data is plotted against the number of UAVs. The companion plots for the figures presented throughout the rest of this section can be found in Appendix C as well.

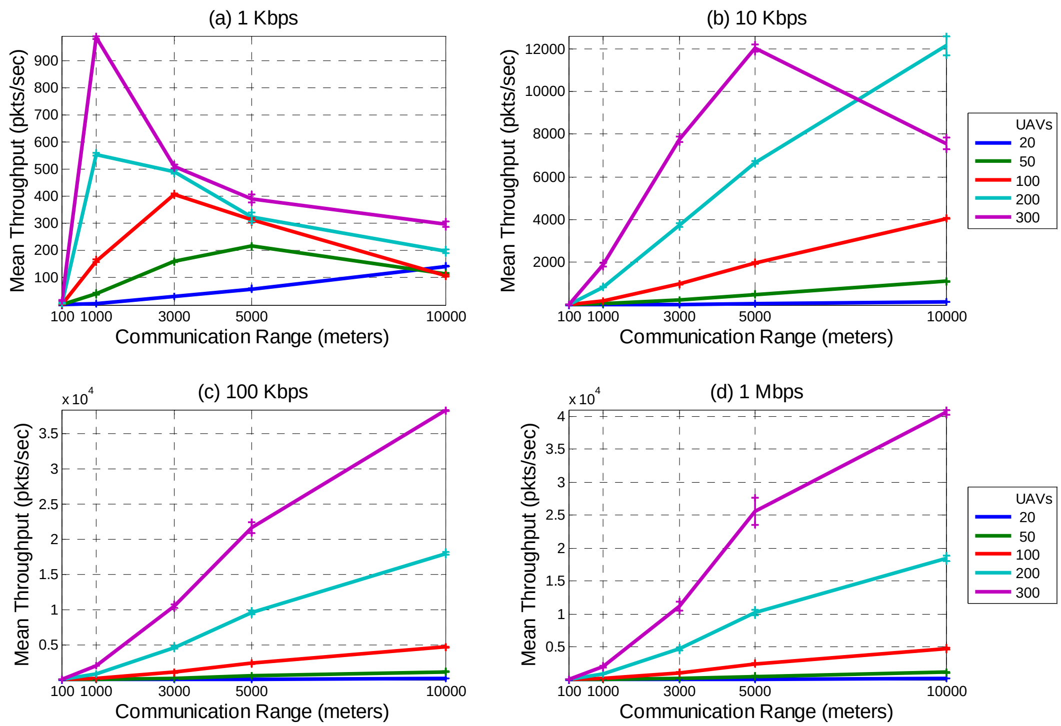

Figure 22. Mean Throughput versus Communication Range

The underlying assumptions of normally distributed errors and constant standard deviation for the ANOVA are not met by the throughput data. The data also does not appear to follow a linear trend. Though a log transformation brings the data closer to meeting the assumptions, it is still not enough to meet the underlying assumptions. Table 7, then, provides the percentage of variation due to each of the factors, interactions, and error. In the table, no one factor or interaction stands out more than the others. However, the percent of variation due to error is very small compared to the factors and interactions. Had the data met the assumptions for the ANOVA, all factors and interactions would have been significant with respect to error.

Table 7. Percent Variation for Throughput

Source	DF	Adj SS	% Variation
Number of UAVs	4	11887330272	23.73
Data Rate	3	3629043796	7.245
Communication Range	4	8041916731	16.06
Number of UAVs*Data Rate	12	5826728176	11.63
Number of UAVs*Communication Range	16	9524673434	19.02
Data Rate*Communication Range	12	4108894572	8.203
Number of UAVs*Data Rate*Communication Range	48	6903338149	13.78
Error	900	169384942	0.003
Total	999	50091310072	

4.2.3.3 Mean Network Delay

Figure 23 presents the average delay experienced by packets across the entire network. Careful inspection of the graphs shows that the increase in data rate from 1 Kbps to 10 Kbps causes a reduction in delay by a factor of approximately 1000. Similarly between the 10 Kbps and 100 Kbps levels the delay is diminished by a factor close to 100 and between the 100 Kbps and 1 Mbps the levels are diminished by a factor of 10. Because of this trend, it is likely that the data rate, communication range, and their interaction explain most of the variation in the network delay. Again the data did not satisfy the assumptions for an ANOVA. The residuals were not normally distributed and the deviations from the mean were not constant. Though a quasi-logarithmic relationship seems to exist between delay and data rate, a logarithmic transformation of the delay did not yield a valid ANOVA. Other common transformations also failed to produce data that could be used in a valid ANOVA.

Table 8 lists the percent of variation due to each of the factors and interactions. As predicted the data rate (22%), communication range (9%), and interaction between data rate and communication range (28%) account for a large portion of the variation in the delay. The delay usually increases quickly once the network becomes congested. Thus, it makes sense that the initial increase in delay in Figure 23(a) occurs at nearly the same locations as the peaks in throughput seen in Figure 22(a). This trend holds for the 10 Kbps level, which can be seen in Figure 23(b) and Figure 22(b) for the 300 UAV case.

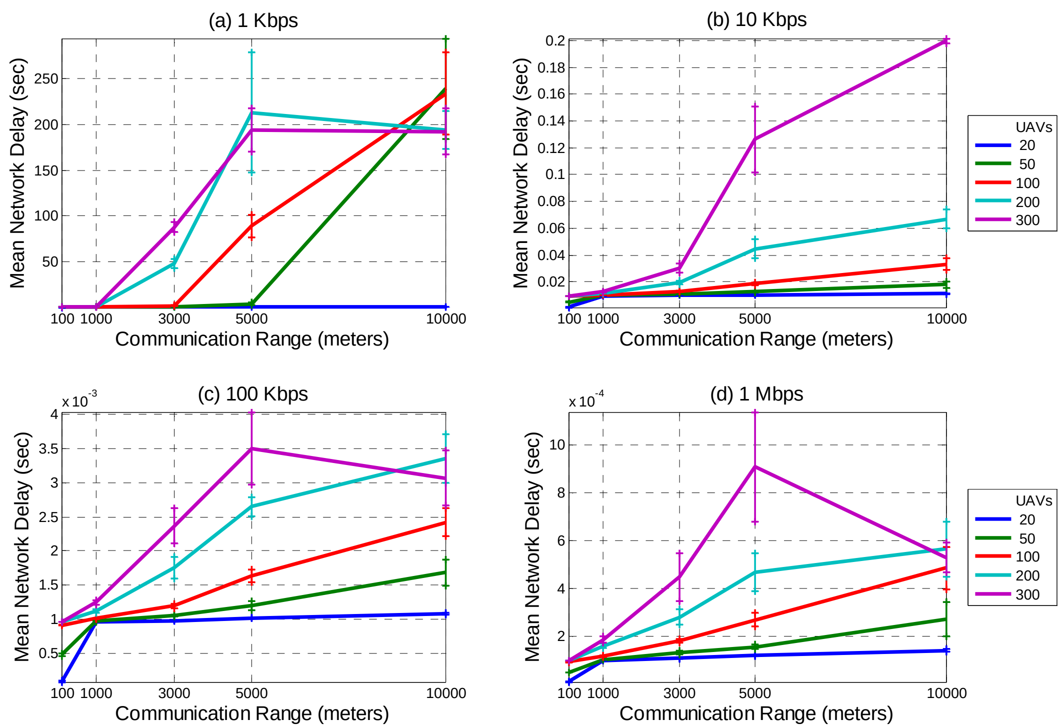

Figure 23. Mean Network Delay versus Communication Range

Table 8. Percent Variation for Delay

Source	DF	Adj SS	% Variation
Number of UAVs	4	73930	2.506
Data Rate	3	673641	22.84
Communication Range	4	279682	9.482
Number of UAVs*Data Rate	12	221476	7.509
Number of UAVs*Communication Range	16	140591	4.767
Data Rate*Communication Range	12	838276	28.42
Number of UAVs*Data Rate*Communication Range	48	421615	14.29
Error	900	300343	10.18
Total	999	2949555	

4.2.3.4 Mean Collision Rate

Figure 24 presents the mean collision rate for the communication network. At first, it appears that all the collisions are primarily related to the communication range and increase in an exponential fashion. If the same figure was plotted using a log-log scale the curves shown appear as straight lines confirming that relationship of the log of the collision rate seems to be linearly related to the log of the communication range. This is shown in Figure 25; however, due to the existence of zeros in the response, some of the points are not plotted in Figure 25. Though the data in Figure 25 would definitely meet the ANOVA criterion that the response has a linear relationship, the residuals do not meet the criteria of being normally distributed or having constant deviations from the mean.

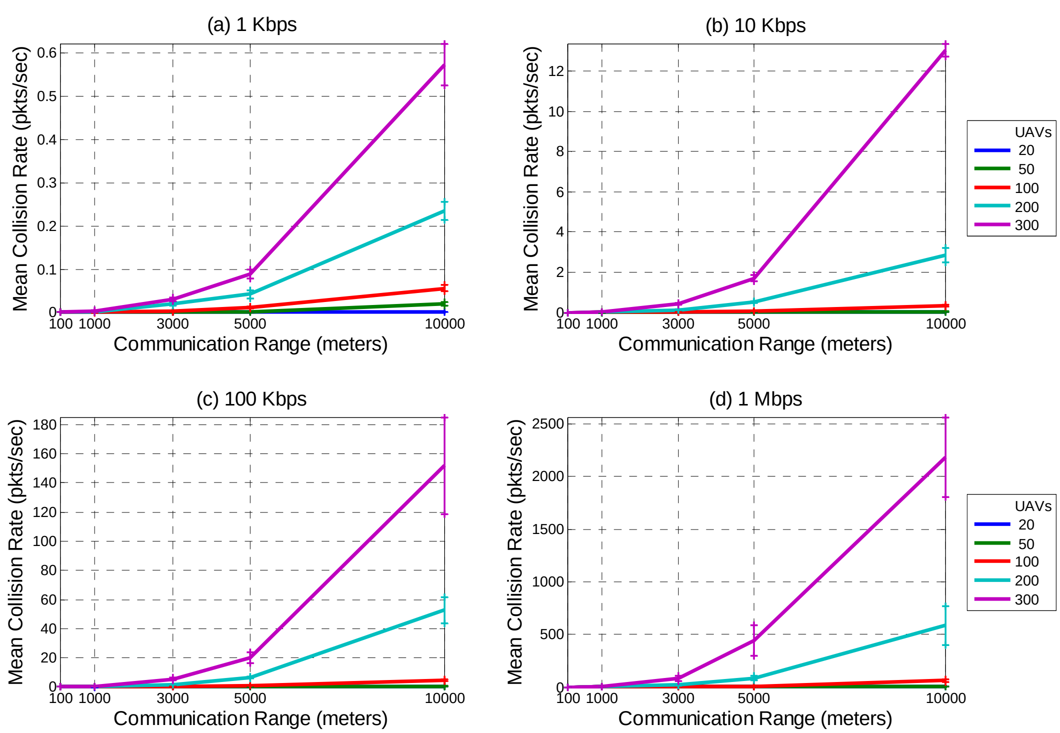

Figure 24. Mean Collision Rate versus Communication Range

Though the collision rate appears to be closely tied to the communication rate, many of the changes in levels in UAVs and data rate are significant. Table 9 shows the percent of variation for the mean collision rate. From the table it can be seen that no one factor accounts for more variations than the others, while the interaction of all three factors (32%) accounts for the largest percent of variation.

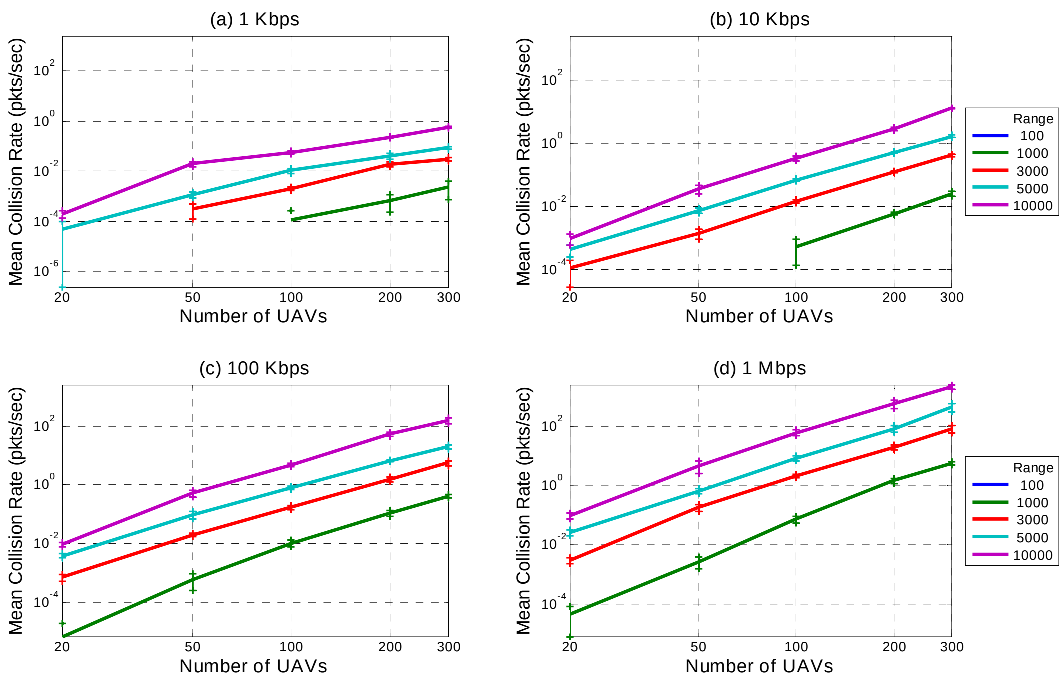

Figure 25. Mean Collision Rate versus Communication Range on a Log-Log Scale

Table 9. Percent Variation for Collision Rate

Source	DF	Adj SS	% Variation
Number of UAVs	4	3120603	5.417
Data Rate	3	3474964	6.033
Communication Range	4	3456714	6.001
Number of UAVs*Data Rate	12	7808420	13.56
Number of UAVs*Communication Range	16	7468656	12.97
Data Rate*Communication Range	12	8453560	14.68
Number of UAVs*Data Rate*Communication Range	48	18462949	32.05
Error	900	5356640	9.299
Total	999	57602506	

4.3 Overall Analysis

The first conclusion to be drawn is that the hypothesis stated in Section 3.1 is not supported by the simulated data. Specifically, it was hypothesized that the number of UAVs, communication range, and data rate would impact the overall search time. The data presented in Section 4.2.1 does not support the claim that data rate is a significant factor in the search time response. Though communication range is a significant factor with respect to error, it explains only a small portion of the variation in search time.

In a previous study using the same UAV models, it was seen that number of UAVs and communication range had an impact on the search time (data rate was not included as a factor) [MMP06]. However, the communication range played a greater role and the best search time was at shorter ranges. One of the major differences in this work was locating all mobile intermittent ground targets present in the simulation defined the underlying criteria for the search time, as opposed to searching all of the cells in the search area. The fact that targets were present caused the UAVs to transition to approaching and orbiting the targets, which temporarily changes the UAV's behavior. It is likely that the presence of the targets is responsible for the differences between the two studies.

The endurance (maximum flight time) of the Desert Hawk UAV, modeled in this study, is approximately 60 to 90 minutes [Jan05]. Assuming that the UAVs are not going to be recharged and launched again, the endurance of the UAVs is important. Using the more conservative 60 minute estimate along with Equation (4.4), a mission to locate approximately 25 ground targets in a 100 square kilometer area should employ 55 UAVs.

This equates to approximately one UAV for every two square kilometers of the search area or two UAVs per ground target. Solving Equation (4.4) for the search time and taking the derivative with respect to the number of UAVs yields:

$$\frac{dT_S}{dN_{UAVs}} = -353602.66 \cdot \left(N_{UAVs}\right)^{-2.1211} \tag{4.5}$$

From this equation when 50 UAVs are engaged in searching, each additional UAV reduces the search time by approximately 74 seconds. At 100 UAVs, the gain has decreased to 20 seconds for an additional UAV, while at 200 UAVs the gain has diminished to 4 seconds and at 300 UAVs the gain is about 2 seconds. Without considering the economic cost of each UAV, it is not useful to add many more UAVs over 100, as each UAV has little impact on the overall search time. If the costs of the UAV are considered, even 55 UAVs is of considerable cost. It should be noted that this analysis is restricted to the parameters of this search space such as area size and area subdivision size.

Because the data rate and communication range do not greatly affect the time it takes to search the area, it is questionable as to whether or not the communication between the UAVs is necessary. When the 1 Kbps data rate is used, the network becomes highly congested and the average delay is about 3 to 4 minutes. In contrast, when the 1 Mbps data rate is used, little network congestion and delay occur. These two cases equate to nearly no communication in the first example to almost complete communication in the second, yet this does not affect overall search time. A similar statement could be made that the 100 meter communication also allows for almost no communication while the 10,000 meter communication range supports nearly complete communication. The fact that these communication contrasts minimally affect the search time would suggest that communication between nodes is either not entirely necessary or that the data being shared is not as helpful to the efficiency of the search algorithm as the communication could be.

Since a UAV tends to travel in a straight line, the sharing of local information may not be as beneficial as sharing information on a more global level. The search redundancy data collected would seem to support this statement as the search redundancy increases for communication range up to the 5,000 meter communication range. But, when the 10,000 meter communication range is reached, the search algorithm becomes much more efficient. Because the UAVs tend to travel in straight paths they tend to travel to relatively distant areas. When shorter communication ranges are used, the UAV is not aware of which areas have been searched recently when arriving at newer territory. This global search stage redundancy increase, coupled with the approach and orbit redundancy increase discussed in Section 4.2.2, are likely explanations for the trends seen in the search redundancy.

If the search redundancy at the 3,000 meter communication range and 5,000 meter communication range could be decreased in some manner it is likely that the overall search times at these ranges would outperform the 10,000 meter case. Because little difference in search time exists between these cases, yet the 3,000 and 5,000 meter range shows much greater redundancy than the 10,000 meter communication range, it stands to reason that UAVs could search an area faster if they did not revisit cells as often.

Part of the goal of this study was to investigate the communication protocol's scalability. Based on the data collected it appears that the protocol will scale relatively well, assuming that the data rate is sufficient. For the particular scenario studied in this experiment, the 10 Kbps data rate is sufficient to handle up to 200 UAVs. However, a margin of safety should be added so that the protocol can scale. Thus, in an actual implementation it would make sense to use a data rate of no less than 100 Kbps. This rate is not unreasonable for the Link16 protocol currently used by the Department of Defense [Hob05]. However, future systems such as the Joint Tactical Radio System (JTRS) are supposed to operate close to 2 Mbps [Hob05], which allows for plenty of bandwidth and a large number of UAVs.

Based upon the trends seen in the amount of data sent, it is likely that a communication network operating at 100 Kbps will be able to support roughly 1,200 UAVs at the 10,000 meter communication range. At the 1 Mbps data rate, the network can support roughly 6,000 UAVs at the 10,000 meter communication range. These numbers are obtained using Equation (4.3) with a packet generation rate of 0.7. This packet generation rate is larger than those observed, however, with a larger number of UAVs, the amount of traffic generated by flooding when a target is located should be much greater.

One item of concern regarding the protocol's scalability is the amount of delay in the network. Large average delays (greater than one second) were only observed in the 1 Kbps data rate. However, the 1 Kbps data rate also represents a situation where the network is seriously over-utilized. Once packets have become very old, they are of less value than current packets. Should the network protocol observe large delays, it would be best if those packets which have been queued longer were 1) given a lower priority as compared to newly arrived packets, or 2) were not transmitted at all. This would decrease the delay for packets that are more likely to impact the search algorithm. If the older packets were discarded, it is likely that the maximum number of UAVs the network could support would be increased. In the 1 Kbps case this increase may only be one or two additional UAVs, but at larger data rates it may be that many more UAVs can be supported.

4.4 Summary

This chapter analyzes the data collected from the simulation scenarios. A validation of the modified IEEE 802.11b network performance, when no targets are present is discussed. The results collected are analyzed individually and as a whole. The applicability of the data collected is also discussed.

V. Conclusions and Recommendations

This chapter contains the conclusions based on this research. Section 5.1 presents a summary of the conclusions drawn from the results. The significance of this research is presented in Section 5.2, while Section 5.3 discusses several recommendations for future research. Section 5.4 provides a brief summary of this chapter.

5.1 Conclusions of Research

The results show that the communication between UAVs has almost no impact on the time it takes to locate all of the ground targets. The data rate was found to be insignificant with respect to experimental errors, while the communication range, which was a significant factor with respect to error, only accounted for one percent of variation in the log search time. The search time was determined to be most affected by the number of UAVs engaged in searching the area. A fairly linear relationship between the log search time and log number of UAVs was discovered. The communication between nodes has little effect, which may indicate the items communicated are of little benefit to the search algorithm.

Though the amount of communication did not impact the search time, it did impact the search redundancy. Search redundancy was generally lower at the 100 meter, 1,000 meter and 10,000 meter ranges, while at the 3,000 and 5,000 meter ranges the search redundancy increased. If the redundancy could be decreased at the intermediate ranges, it may be possible to decrease the search time where the redundancy is highest, which occurs at the 3,000 and 5,000 meter ranges and 100 Kbps and 1 Mbps data rates.

When the network became highly congested (as in the 1 Kbps case), the average delay became very large. In terms of the search application, a small delay is better as newer information is more important than older information. The delay is most likely the result of packets being queued by the Media Access Control (MAC) Layer prior to transmission. Considering the amount of traffic generated, network congestion, and realistic data rates available, using a network with a 100 Kbps or faster data rate should allow for minimal congestion and for a greater degree of scalability.

5.2 Significance of Research

Though not a primary focus of this research, the implementation of the search algorithm used for these experiments is the first that models each UAV as a self-contained entity. The results show that the proposed four-stage search algorithm can be implemented as a distributed algorithm and can perform its intended function. This alone adds to the body of research in [PaM03], [PYT05], and [YPH06] and adds to the development of the search algorithm by demonstrating its effectiveness.

In conjunction with making the algorithm truly distributed, it was necessary to design those parts of the communication protocol that were not previously considered and improve upon the design that previously existed. Specifically, this included designing the part of the protocol by which the UAVs shared information about the targets, as well as including next destination information instead of current location for global search communications. Though these topics are not presented in the main body of the text, they are presented as a part of Appendix A.

This research added realistic wireless communication between nodes to share information between UAVs. The effect of wireless communications on the search algorithm has been investigated. In an actual implementation, the number of UAVs, data rate, and communication range affect the performance of the system is critical to the creation of a good overall design.

Another contribution of this research has been to enhance the research being performed by the UAV Research Group at the U.S. Air Force Academy. Building a UAV research bridge between the Air Force Institute of Technology and the Air Force Academy is a secondary purpose of this research. The contribution that this research makes is a start to creating this bridge.

5.3 Recommendations for Future Research

Since the search time is not greatly affected by the amount of communication, it is possible that a better communication protocol exists. During the global search state, instead of transmitting what area the UAV will search next, it may be more beneficial to send information about the current cell as well as cells in the surrounding area. This could create a more cohesive area map within each UAV. When sending information about targets, it might be useful to include information about how many UAVs may already be orbiting the target (from the perspective of the sender). UAVs in orbit around a target could coordinate their communications. Currently, the UAVs send their most recent data point every half second after entering orbit around the target. Once three UAVs are orbiting a target, they could communicate using less power to allow more communication to occur across the network. This could improve the scalability of the protocol. Currently, a network flood is used to communicate the fact that a target has been located. However, if regular UAV communications included the UAV's estimate of the number of targets remaining, this information would eventually diffuse through the network and flooding the network solely to communicate this information would be unnecessary. In such a protocol, it would eventually be necessary for the sensing UAVs to communicate a target's location information to a platform that could perform the necessary action to deal with the target. Though flooding could be used, a routing algorithm that consumed fewer network communication resources would likely produce better results.

Another avenue for future research involves the size and shape of the search area and associated underlying grid. This research used a 100 square kilometer search area subdivided into squares. However, the effect that different search area shapes and sizes has on search time is undetermined. Exploration of these parameters could lead to a generalization of Equation (4.4) and Equation (4.5) to more diverse search areas. Often, hexagonal grids are used in simulation because hexagons provide a better approximation of a circle. Since the sensors on the UAVs are assumed to have a circular range associated with them, a hexagonal grid may be better.

In conjunction with the search area, the number of targets or the UAV-to-target ratio could impact the performance of the system. This research used a fixed number of ground targets. It would be useful to determine the effect that the density of targets has on the search algorithm. For different numbers of targets, different initial UAV deployments may yield a lower overall search time to destroy the targets. The deployment method alone may be a topic of interest as well.

The model fidelity could also be improved. Currently, very few realistic flight characteristics are modeled and in this work it was possible, however unlikely, for UAVs to instantaneously turn 180 degrees. The presence of obstacles and terrain could be implemented within OPNET Modeler through the use of Digital Elevation Model (DEM) and Digital Terrain Elevation Data (DTED). It would be necessary to implement or design an obstacle avoidance algorithm that will work in conjunction with the search algorithm. OPNET Modeler also provides a framework for co-simulation with other programs as well as hardware. It could be useful to let the search algorithm implemented in OPNET choose destination waypoints and pass these waypoints to an external flight simulator that would in turn model more realistic UAV movement. The current UAV position would have to be reported by the flight simulator to OPNET for use by the search algorithm as well as for use in modeling the communication between nodes. In terms of the system diagram presented in Figure 7, on page 32, all that would be different is that the UAV navigation system is implemented in a flight simulator rather than in OPNET.

Because the communication network could support a greater traffic load at the larger 100 Kbps and 1 Mbps data rates, it should be possible to integrate this application of a search swarm with other UAV swarm applications. For instance, an implementation of a communication swarm, where the network is used to extend communications could be implemented simultaneously with the search algorithm. Depending on the priority of each task, the UAVs could position themselves to aid in the search for targets, the creation of a dynamic communication backbone, or a combination thereof.

In addition, integrating the UAVs using the search algorithm with an existing radar network may also be advantageous. When a target is detected by the radar system, UAVs could be tasked to investigate the target and provide more information than what the radar could alone. In addition, the UAVs could search the area as well, and could potentially find targets not detected by the radar.

5.4 Summary

This chapter presents the overall conclusions that are drawn from the results. Also discussed are several aspects of the significance of this research. Several recommendations for future research are given as well.

Appendix A – Implementation Details

This appendix explains the implementation details used in the experiments that are discussed in the main body of this document. Due to the length of this material, it has been included in an appendix so that it does not inundate the reader and obscure the purpose of the main portion of the paper, which is to outline the experiment and discuss the results.

This appendix is divided into multiple sections. Section A.1 discusses the search algorithm in detail. Section A.2 discusses the necessary modifications that had to be made to the OPNET implementation of the IEEE 802.11b protocol for correct operation. Also discussed in this section are the sub-parameters used to create the communication ranges used in the experiment. Section A.3 outlines the packet formats used for communication between UAVs. Section A.4 discusses the operation of the ground targets, while Section A.5 discusses the operation of the UAVs. Section A.6 discusses the operation of necessary global processes as well as the global parameters that are available to the user.

A.1 Search Algorithm

As mentioned in Chapter II, the studies [YPH06] and [PYT05] build off of the previous study [PaM03] and define a four-stage search process. These four stages are:

1) Global search for targets,
2) Approach a detected target,
3) Orbit and refine a target location, and
4) Local search for a lost mobile target.

In the first stage, the four universal search rules from the study [PaM03] are adapted into a slightly different rule set more appropriate for UAVs. Given the current location and direction of travel of a UAV in stage 1, the rule set defines that the UAV should try to:

1) Seek out a position that was searched least recently,
2) Seek out a position that lies farthest from the current locations of neighboring UAVs and the search area boundaries, and
3) Travel in the same direction that the UAV is currently traveling (for fuel efficiency).

These three rules are embodied in Equation (A.1), which is used to produce a search score, S, for a finite set of potential next destinations. It should be noted that Equation (A.1) is not exactly as presented in [PYT05] and [YPH06], as the form shown here includes recent updates and fixes a typographical error. Since the search area is made up of a rectangular grid, the UAVs consider the eight surrounding cells as potential destinations, unless the UAV is next to a corner or edge, in which case fewer cells are considered for the next destination. The adjacent cell with the lowest search score is the next cell the UAV will visit and scan. Each of the three multiplicative terms in the equation corresponds to one of the three rules listed. The first term, *H*, represents a numerical value based upon how long ago the cell under consideration was last searched. The value of *H* used by the UAVs is the sum of one and the simulation time that the cell was last known to have been searched by any UAV. The range of possible values begins at one. For example, the last search time of cells that have not been searched is considered to be zero and the value of one is used for *H*. The reason that the sum is used is to prevent an *H* value of zero from cancelling the contributions of the other terms in Equation (A.1). As time progresses the cells that have not been searched become increasingly likely to be chosen as the next destination of a UAV.

$$S = H\left(\sum_i \frac{1}{D_i} + \sum_j \frac{1}{D_j}\right)\sqrt{1+\frac{|\phi|}{\pi/p}} \tag{A.1}$$

The next term consists of D_i and D_j where D_i is the distance from the center of the cell under consideration to the "i"-th UAV and D_j is the distance from the center of the cell to the search area boundary "j". The last term in the equation consists of ϕ, which is the turn angle required for the destination cell, and *p*, which represents the number of discrete points used in determining the turn angle.

In stage 1, the UAVs move from cell to cell within the search area grid. In the experiments performed, the cells were squares with 50 meter sides. In the square search area with 10 km sides used in the experiment, the grid is composed of 200 x 200 cells. Each UAV keeps a record of the last time that each cell was searched. It is assumed that all UAVs are configured with the same parameters so their grids are of the same size and shape. A UAV considers a cell searched when its current position is within the *central quarter area* of the cell, or when it receives communication from another UAV indicating that a particular cell was searched. In this implementation, the UAV must be able to scan the entire cell once it is in the *central quarter area*; hence, the size of cells and the minimum sensor radius are related. This relationship can be described, for rectangular cells, by

$$R_s = \frac{3}{4}\sqrt{c_x^2 + c_y^2} \tag{A.2}$$

where R_s is the minimum sensor radius, and c_x and c_y are the x and y dimensions, respectively, of the individual cells in the search grid. The *central quarter area* of a generic cell is illustrated with the minimum sensor radius in Figure 26 for clarity. The minimum sensor radius for the cell shown in Figure 26 is about 54 meters.

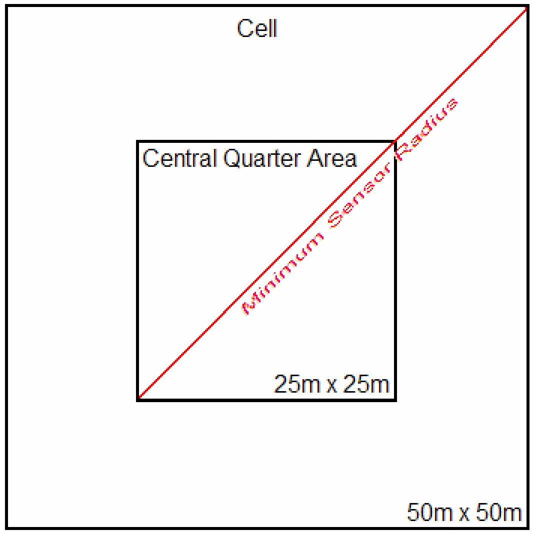

Figure 26. Generic Cell in Search Area Grid

When a UAV detects a target, it immediately moves from stage 1 to stage 3 and sends a request to the other UAVs for assistance in finding the exact location of the target through a form of triangulation. If a UAV's cooperation is requested, the UAV first assesses the cost of helping in triangulation versus continuing the global search. If the distance is too great or there are already three UAVs assisting, then the UAV continues to search its immediate area and ignores the request for assistance. The decision to help is made by each UAV receiving a request through the use of Equation (A.3). If the resulting value of C is negative then the UAV will enter stage 2 and approach the target, otherwise the UAV remains in stage 1 and continues the global search for targets.

$$C = w_1 \frac{D}{D_{\max}} - w_2 \left(n - s\right) + w_3 \left(m - p\right) \qquad \text{(A.3)}$$

In Equation (A.3), the w_x terms are weighting factors that can influence the UAVs' decision making process. In this implementation, the following values were chosen because these values were used in the previous study [YPH06] and produce appropriate behavior: $w_1 = 2$, $w_2 = 1.5$, and $w_3 = 0.5$. The term D is the estimated distance from the UAV considering assisting to the target in question. D_{max} is the maximum straight-line distance in the search space. The maximum straight-line distance is the largest possible distance that a UAV could potentially travel in a straight line. For rectangular search areas, this would be the distance between non-adjacent corners. The term n in the equation is the maximum number of UAVs that should orbit any one target at a time. In previous work, it was determined that this value would be three. The reason that three is chosen has to do with information presented in the paper which shows that for each UAV over three, little additional information is gained when locating the target [YPH06]. The term s is the number of UAVs currently orbiting the target. The term m is the total number of targets estimated to be in the search space and the term p represents the number of those targets that have already been located and eliminated. In this implementation, D_{max}, m and n are known when the simulation begins and D, s, and p are determined based upon communication sent by other UAVs.

A UAV immediately enters stage 3 when it detects a target within its sensor range, if it is in stages 1 or 2. In stage 3, the UAV orbits the target and periodically sends information about the target's position. These periodic messages cause other UAVs to consider entering stage 2 and are used by UAVs orbiting a target to share information and cooperatively locate the target. Because triangulation is used, while in stage 3 the UAVs will speed up or slow down in order to spread themselves out so that better triangulations can be achieved. Though the UAVs speed up and slow down they may not appear to separate because each UAV is only aware of the locations of the others in orbit when the UAVs communicate. It is possible for them to be physically close together, but since the

UAVs are not synchronized, they would communicate at different times and provide appropriate measurements of the target's location. In the implementation used for the experiments, once suitable information was obtained for a UAV to triangulate the target, the target was eliminated and the fact that the target was destroyed is flooded across the network. Once a target is destroyed or cannot be tracked by the UAVs, the UAVs orbiting the target will return to stage 1 and resume the global search. Stage 4 is not implemented due to time constraints.

A.2 Network Adjustments

The IEEE 802.11b standard was designed for a small network where no one node was farther than 300 meters from another node in the network. The implementation of this protocol in OPNET also produces a warning if two nodes on the same wireless network are more than 300 meters apart. However, the IEEE 802.11b standard can be expanded to a few kilometers before problems due to the range limitation occur. The standard as it is written will not work properly for the wireless network implemented in this experiment without some slight modifications. The problems arise because the interframe spaces are not long enough.

The interframe spaces are generally defined based upon the slot time. The slot time is set such that "a station will always be capable of determining if another station has accessed the medium at the beginning of the previous slot" [Bre97]. In simpler terms, the slot time is the largest one-way propagation time possible in the network. For a 15 km, the one way propagation time of an electromagnetic wave is about 50 μs. A distance of 15 km is used to allow for a margin of safety; the actual maximum distance for this experiment is 14.14 km. The short interframe space (SIFS) is generally half the slot time, but can be slightly longer. The distributed interframe space (DIFS) is defined as the sum of the SIFS time and twice the slot time. The SIFS time was chosen to be 28 ms, so that the DIFS time would be 128 ms, which matches the DIFS time discussed in [PaM03]. The values of the SIFS, DIFS, and slot time also happen to be the same as defined in the IEEE 802.11a protocol. The timing modifications are summarized in Table 10.

Table 10. Summary of Timing Modifications Made to IEEE 802.11b Protocol

Parameter	IEEE 802.11b Standard	Modified Values for 10 km
Slot Time	20 μs	50 μs
SIFS	10 μs	28 μs
DIFS	50 μs	128 μs

Another modification made has to do mostly with OPNET's implementation of the IEEE 802.11b protocol. Because the protocol was not designed to operate at a data rate less than 1 Mbps, there are problems with the computation of the physical layer convergence protocol (PLCP) header overhead. Rather than fix the numerous places that would need to be corrected, the PLCP overhead that was simulated was simply eliminated from the packet. Though the PLCP still exists, its size (overhead) is not used in the calculation of frame transmission times.

In order to set the effective communication range, two sub-parameters must be set correctly. These parameters are transmission power and receiver sensitivity. OPNET provides an equation to roughly estimate the communication range [Opn04]. The communication ranges of several combinations, including those used in the experiments are shown in Table 11. The combination of parameters for the approximate 300 meter range is the default combination for the models that use the IEEE 802.11b protocol.

Table 11. Listing of Sub-parameters for Communication Ranges

Receiver Sensitivity (dBm)	Transmission Power (mW)	Theoretical Range (meters)	Approximate Range (meters)
-95	350	10464.878	10000
-95	185	7608.272	7500
-95	85	5157.148	5000
-95	32	3164.282	3000
-95	15	2166.436	2000
-95	10	1768.887	1500
-95	5	1250.792	1000
-91	5	789.197	750
-95	1	559.371	500
-90	1	314.558	300
-87	1	222.690	200
-81	1	111.609	100

A.3 Communication Format

There are two different frame formats used for communication between UAVs. The first of these is used to communicate that a cell has been searched and the other is used to communicate a target’s position or elimination. For ease of discussion, the former shall be called the “search frame format” and the later will be called the “target frame format”. Figure 27 shows the 96-bit format of a search frame while Figure 28 depicts the 144-bit format of a target frame.

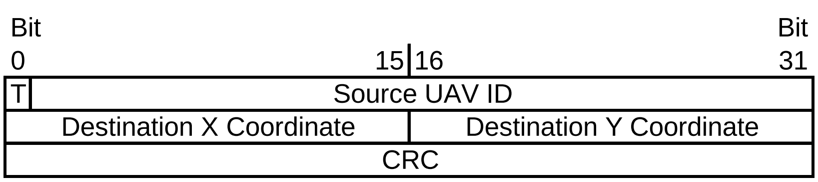

Figure 27. Search Frame Format

Bit 0	15 | 16	Bit 31
T	Target ID	
Source UAV ID		
Current UAV X Coordinate		Current UAV Y Coordinate
Target's Bearing Angle		CRC
CRC (cont)		

Figure 28. Target Frame Format

The search frame format is similar to the format described by Pack and Mullins [PaM03]; however, two differences exist. The first is the presence of the *T* bit field, which is used to differentiate between the two frame formats. The *T* bit is zero for search frames and is one for target frames. It should be noted that this is not the way this is actually implemented. Instead, all UAV identifiers are positive and all target identifiers are sent as a negative value. This method is essentially the equivalent of having a *T* bit field.

The second difference in the search frame format is that instead of sending the current X and Y coordinates, the coordinates of the UAV's next destination are sent. The reason for this is that the second global search rule requires knowledge of the locations of other UAVs in the group. However, UAVs are not aware of the current location of other UAVs and rely upon the last communication received to estimate each other's locations. The last known position of other UAVs is stored and used as an estimate for the true position of the UAVs. Because location estimates are used, it is better to communicate the UAV's next destination when it is chosen. This helps to keep UAVs that are physically close together from inefficiently following the same path and searching the same cells.

The following scenario illustrates this phenomenon. Two UAVs, UAV1 and UAV2, are located to the west and east, respectively, of a position that was last searched a considerable time in the past (such that this factor outweighs the UAV's close proximity to each other). If UAVs transmit their current location instead of their next destination, both UAVs will choose to visit the same location next. Assuming UAV1 completes its scan of the cell first and broadcasts its position, it factors in that the last known position of UAV2 was to the east and based upon the algorithm chooses to go north as its next destination. After UAV2 scans the area and begins to choose its next destination, it considers the last known location of UAV1 to be its current location. To UAV2, any other location is farther from UAV1. All other factors being the same, UAV2 will likely choose the same destination as UAV1 and will likely follow UAV1 from cell to cell. Preliminary testing indicated that once a similar situation occurred, the two UAVs continued to traverse the same locations for a considerable amount of time. This situation is undesirable because the second UAV is inefficiently utilized.

When UAVs broadcast the next destination information, the previous scenario would not arise. Should two UAVs be in the same location, once the first UAV chooses its next destination and broadcasts this information, the second UAV would consider the same location to have been searched as well as be the last known location of its neighbor. These two factors would most likely cause the second UAV to choose a different destination. This does not entirely eliminate UAVs from traveling together, but does make this inefficiency a rare event.

Target frames are only sent by UAVs when the UAV is orbiting a target (stage 3). When a UAV in the global search stage receives a target frame, the UAV evaluates whether it should assist in locating the target or not. If a UAV that is on its way to a target (stage 2) receives a target frame, one of two events will take place. If the target frame has the target identifier for the target the UAV is approaching, the UAV uses this new information to approach the target. If the target identifier is different, then the UAV will approach the target that is the closest to its current position. Regardless of the action taken in stages 1 or 2, the information passed in a target frame is stored by the UAV until the information gets too old. Since the source UAV's identifier is included in the frame, UAVs in stages 1 and 2 can determine how many UAVs are currently orbiting a particular target. UAVs that are orbiting a target (stage 3) send target frames approximately once every half second in order to share their information about the target's location with other UAVs in orbit and to enlist the aid of other UAVs to help refine the target's location.

Once a UAV is able to use triangulation to determine the target's location, it broadcasts this information using a special form of the target frame. This frame is different in that instead of including the UAV's position in the frame, the estimated target position is included in the frame and the bearing angle is set to negative one. Any UAV that receives a target frame with a negative bearing angle will rebroadcast the frame if it has not received such a frame for a particular target before. If rebroadcast, the UAV marks the target as destroyed and will ignore future packets of this type for any target that it knows has been destroyed. This is essentially a network flood and is useful for updating the number of targets remaining for each UAV, which is necessary in the equation for evaluating whether or not to transition to stage 2.

Lastly, it should be noted that a reliable transport mechanism is not used in this application for two reasons: (1) old information is of less value than current information and (2) the loss of a single data frame at a receiver is not likely to have large impact on the overall performance of the distributed search algorithm.

A.4 Mobile Intermittent Targets

The targets are represented by an icon of a Humvee, like the one that can be seen in Figure 30, later in this section. The node model for the mobile targets consists of a single processor. This processor contains two process models - the main process model that controls the mobile behavior of the target and another that controls whether or not the node is currently radiating energy (which determines if the target can be detected by the UAVs). The main process model, shown in Figure 29, will be discussed first followed by a discussion of the child process model.

When the simulation is started, the main process model initializes several items. One of these is to schedule the first *target navigation interrupt* and *target boundary interrupt*. The *target boundary interrupt* is only scheduled if the target is currently within the bounds of the simulation. The object identifier for the global search processor (discussed in Section A.6) is also obtained. The initialization state will also create and invoke the child process model. Once initialization is completed, the process moves to the hidden state.

The hidden state does not mean that the target is not transmitting; rather, it means that the target has not yet been located by a UAV. While in the hidden state, when the process handles a *target navigation interrupt*, a new random speed and bearing are created. Also, a new *target navigation interrupt* is scheduled for a future time, which is determined by the global parameter *target bearing update interval*. The new speed is generated using a uniform random distribution ranging from 10 meters per second to 15 meters per second. The bearing is also generated from a uniform random distribution that ranges across integer values from 0 to 359 degrees.

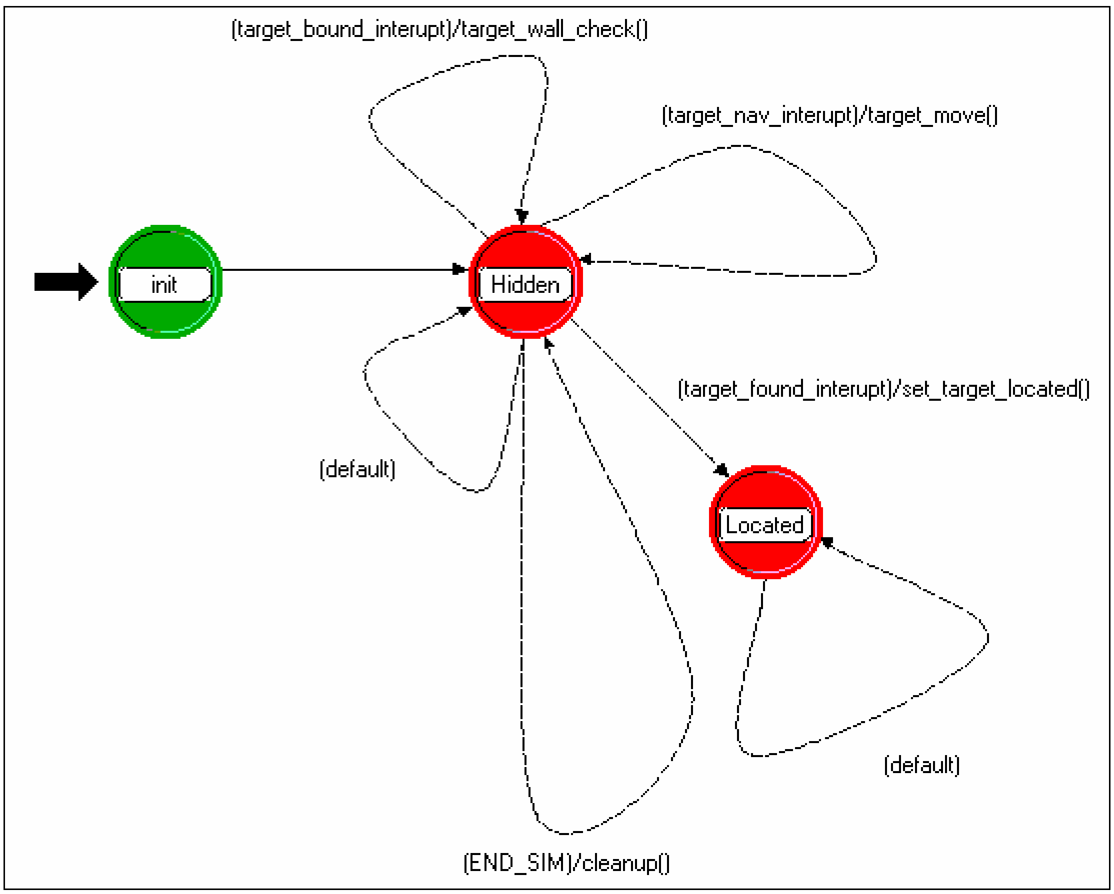

Figure 29. Main Process Model for Mobile Intermittent Targets

When a *target boundary interrupt* occurs while the process is in the hidden state, the current position of the target is compared to the simulation area boundaries. If the target is beyond at least one of those boundaries, then the target's bearing will be modified so that the target will stay within the boundaries. The target's behavior at the boundary is such that the target's trajectory will obey the rules of a simple inelastic collision, where the angle of incidence, ϕ_i, is equal to the angle of reflection, ϕ_r. A pictorial example of this is given in Figure 30. The *target boundary interrupt* is rescheduled so that the target's position will be checked and modified if necessary once a second during the simulation.

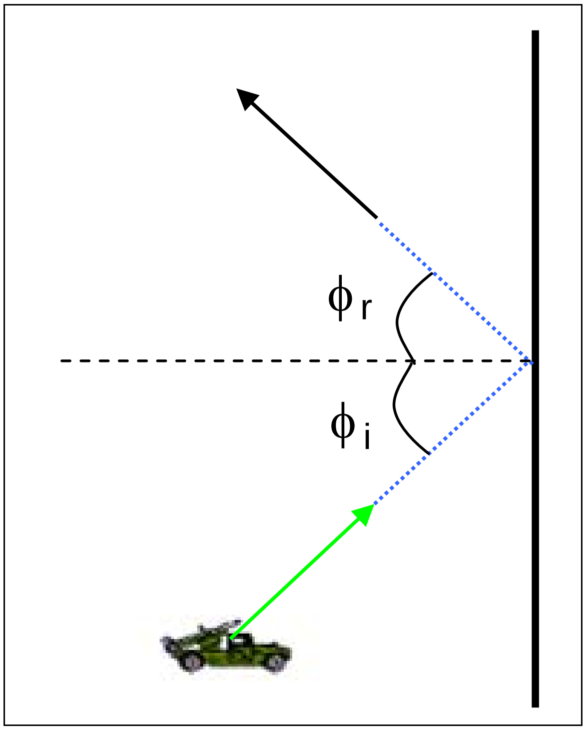

Figure 30. Example of a Target's Trajectory Undergoing
an Inelastic Collision with an Area Boundary

Should the simulation end before the target has been located, the main process will execute the cleanup() function, which serves to terminate the child process. When a UAV locates the target, the UAV schedules the *target found interrupt*; the target does not schedule this interrupt itself. When the *target found interrupt* occurs, the target's speed is set to zero, the child process is terminated, and the node's condition is set to disabled.

Setting the node's condition to disabled causes all processes running within the node to be terminated. Because the UAVs check the condition of the target when it is in range of their sensors, the UAVs will ignore the target when it is disabled.

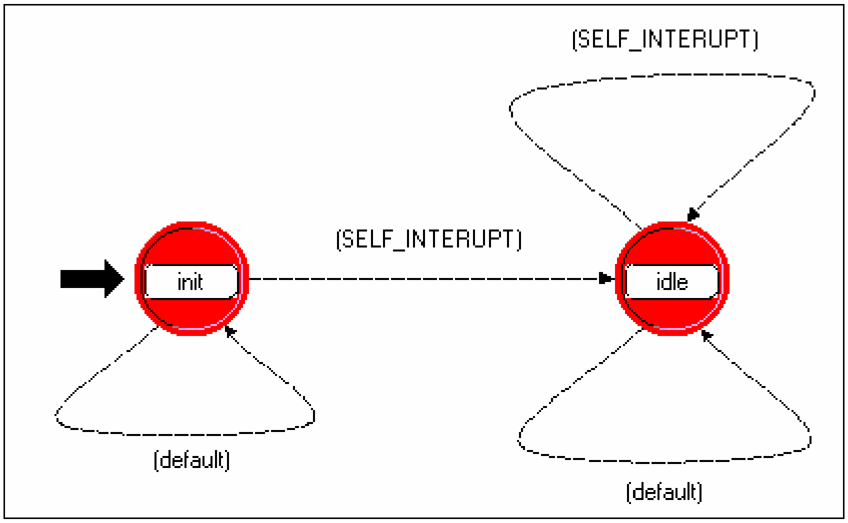

Figure 31. Child Process Model for Mobile Intermittent Targets

Figure 31 shows the behavior of the child process model. When the child process is first invoked, it will perform necessary initialization and if target intermittency is globally enabled, the process will turn off the transmitting attribute and schedule a self interrupt and transition to the idle state. If target intermittency is not enabled, the process will remain in the initialization state with the transmitting attribute on until the process is destroyed. The transmitting attribute is a node attribute that can be accessed by the UAVs. If the attribute is on (and the target is not disabled), the UAVs will be able to detect the target. If the attribute is off, the UAVs will not be able to detect the target. While in the idle state, when a self interrupt occurs, the value of the transmitting attribute is toggled between the values on and off. The self interrupt is also rescheduled depending on the current state of the transmitting attribute. These times are randomly generated using *on time* and *off time* global parameters. The scheduling is done such that the target's transmitting attribute will be on for a uniformly random time between 1/2 the value of *on time* and 3/2 the value of *on time*. Similarly, the target's transmitting attribute will be off for a uniformly random time between 1/2 the value of *off time* and 3/2 the value of *off time*. Each time the transmitting attribute is toggled, an interrupt is generated in the *search configuration node*, described in section A.6, which causes the target's icon to change. The target's icon is a humvee, as previously stated, when the transmitting attribute is on. The icon is changed to a yellow circle when the transmitting attribute is off.

A.5 UAV Models

A generic diagram to describe the search algorithm as it is implemented in each UAV is presented in Figure 32. In the figure the blue dotted lines represent that portion of the search algorithm that has not currently been implemented. In the figure, red denotes those actions that are preformed instead of the unimplemented blue sections and should be removed when the blue sections are implemented. Figure 32 is meant to help summarize Section A.1 as it pertains to this section as well as to help alleviate any confusion that may arise as many low-level detains are described in this section.

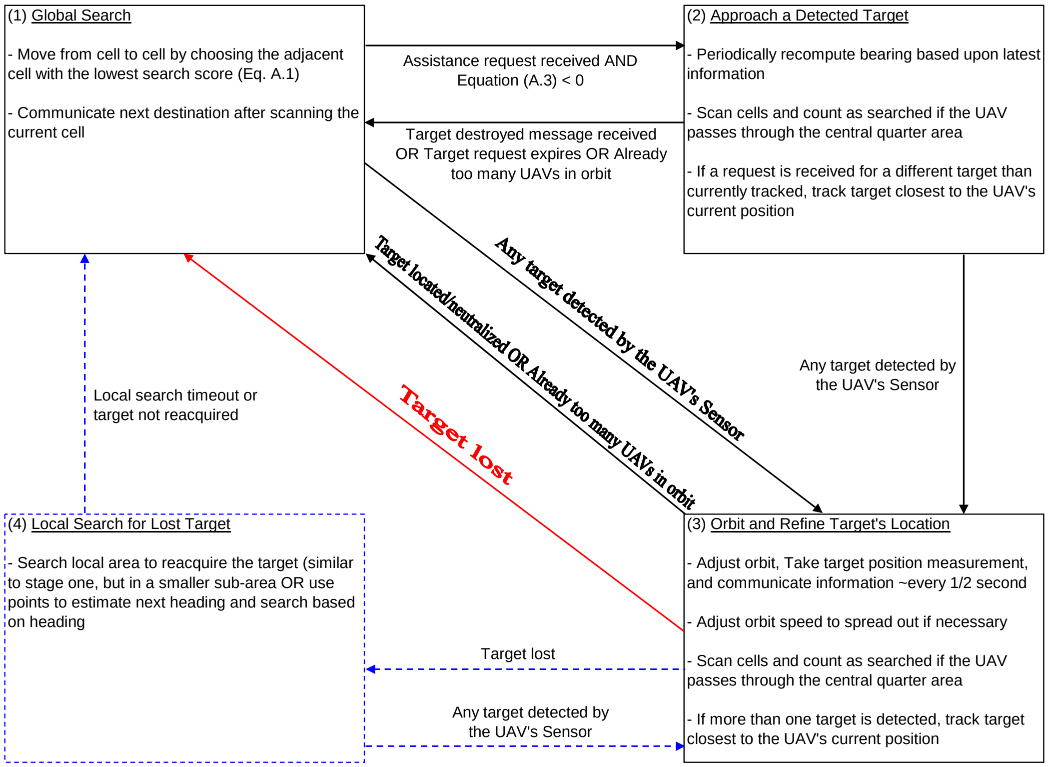

Figure 32. Generic State Diagram of the Search Algorithm Implemented in Each UAV

The node model for each UAV is shown in Figure 33. There are two main sections to the node model. The first section is slightly different from the base OPNET model used to construct the UAV model and is shown in the green dashed box. These modules are responsible for the wireless communication between UAVs. In the original OPNET *wlan_station_adv* model, this part of the node model receives packets from a packet source and delivers packets to a packet sink. The Mobility and Mapping CPUs supply and receive packets, but actually process and react to the information contained in the packets. Together with the Mobility Support CPU, these modules form the second section of the node model, which was built in order to implement the search algorithm.

Though it is not evident in Figure 33, the Mobility CPU and Mapping CPU are linked in that these modules share data and communicate with each other via interrupts.

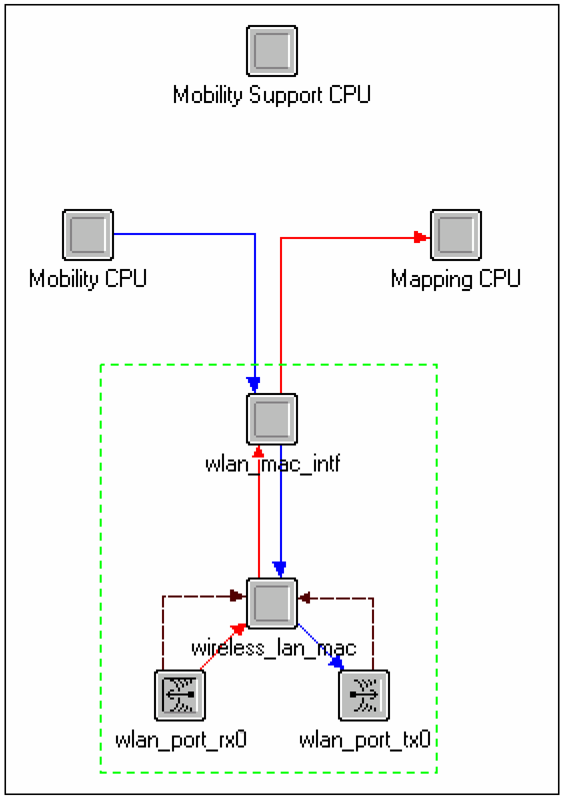

Figure 33. UAV Node Model

The *wlan_mac_intf* module acts as an interface between the MAC layer modules and higher layer modules. This module adds the necessary data structures and formatting to packets coming from the modules outside the green box in Figure 33. The interface also strips the added constructs from packets when they are sent to the higher layer modules (outside the green box). The *wireless_lan_mac* module contains OPNET's implementation of the IEEE 802.11 collection of protocols. The necessary modifications to the network models discussed in Section A.2 were made to the process within this module. When a packet is received from the higher layer, the packet is encapsulated into a frame and the CRC is added, so that it conforms to the frame formats discussed in Section A.3.

The next several paragraphs discuss the Mapping CPU, Mobility CPU, and Mobility Support CPU, in that order. The reason for this is that the operation and interaction of Mapping CPU and Mobility CPU are more likely to be understood. The Mobility Support CPU has become a nearly vestigial component. The functions that it performs are no longer necessary, but were necessary in the development of the node model. Because it is of limited importance, it is discussed last.

The main function of the Mapping CPU, as its name may suggest, is to maintain the UAV's virtual map of the search area. It also is responsible for keeping track of the last known locations of other UAVs and processing incoming communications from other UAVs. The Mapping CPU contains a single process model, which is presented in Figure 34. During the initialization stage, the process allocates and initializes the search area map and neighbor location table. Once these tables are initialized, the process immediately sends a remote interrupt to the Mobility CPU. The reason behind this is that part of the initialization of the Mobility CPU requires that the search area map and neighbor location table exist and must wait until these structures have been created. After initialization, the process waits for update information to arrive in one of two ways. The first of these occurs when a cell in the search area is scanned, in which case, the Mobility CPU will send a remote interrupt to the Mapping CPU. The Mapping CPU's process model reacts by marking the cell the UAV is in currently as scanned by entering the current time into its search area map. The second update method occurs when the Mapping CPU receives a packet from another UAV through the wireless communications network. When a position packet is received, the search area map is updated using the position from the packet and the time that the packet was received. Also, the neighbor location table is updated with the source UAV identifier and position from the packet as well as the time that the packet was received.

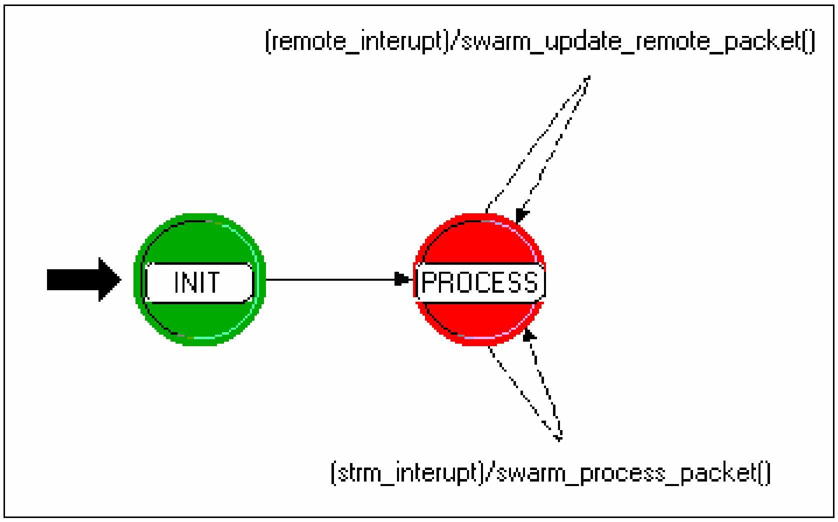

Figure 34. UAV Mapping CPU Process Model

When a target packet, that does not indicate that a target was located, is received the process evaluates whether the UAV should assist the sending UAV. If the UAV is currently in phase 1 and is supposed to enter phase 2 based on the information in the packet, the Mapping CPU sends an interrupt to the Mobility CPU to signal the Mobility CPU to make the transition to stage 2. Regardless of whether the interrupt is sent, the neighbor location table is updated and the target information table (contained in the Mobility CPU) is also updated. If the target packet indicates that the target has been located, the Mapping CPU checks the target information table to determine whether or not the target was marked located or not. If the target was not previously located, the target is marked located and the packet is forwarded. If the target was already known to be located, the packet is destroyed and not forwarded.

Because the Mapping CPU does not have an outgoing packet stream to the MAC interface, a special OPNET kernel procedure is used to deliver a forwarded target packet to the interface. To the interface, it appears that the packet was sent from the Mobility CPU. The packet-sent statistics kept by the Mobility CPU are written when a packet is sent in this manner. The Mapping CPU keeps statistics on the packets it receives and destroys packets after they are processed.

The process model for the Mobility CPU is shown in Figure 35. During initialization, the Mobility CPU uses the known number of targets to dynamically allocate and initialize the target information table. As previously discussed, the process must wait for the Mapping CPU to send a remote interrupt. After that occurs the process is able to determine the UAV's destination. When initialization is complete the process transitions to the phase 1 state. The phase 1, phase 2, and phase 3 states correspond with each of the three search stages that have been implemented (global search, approach target, and orbit and refine the target's position). This process model is responsible for keeping statistics in regards to the amount of network traffic sent.

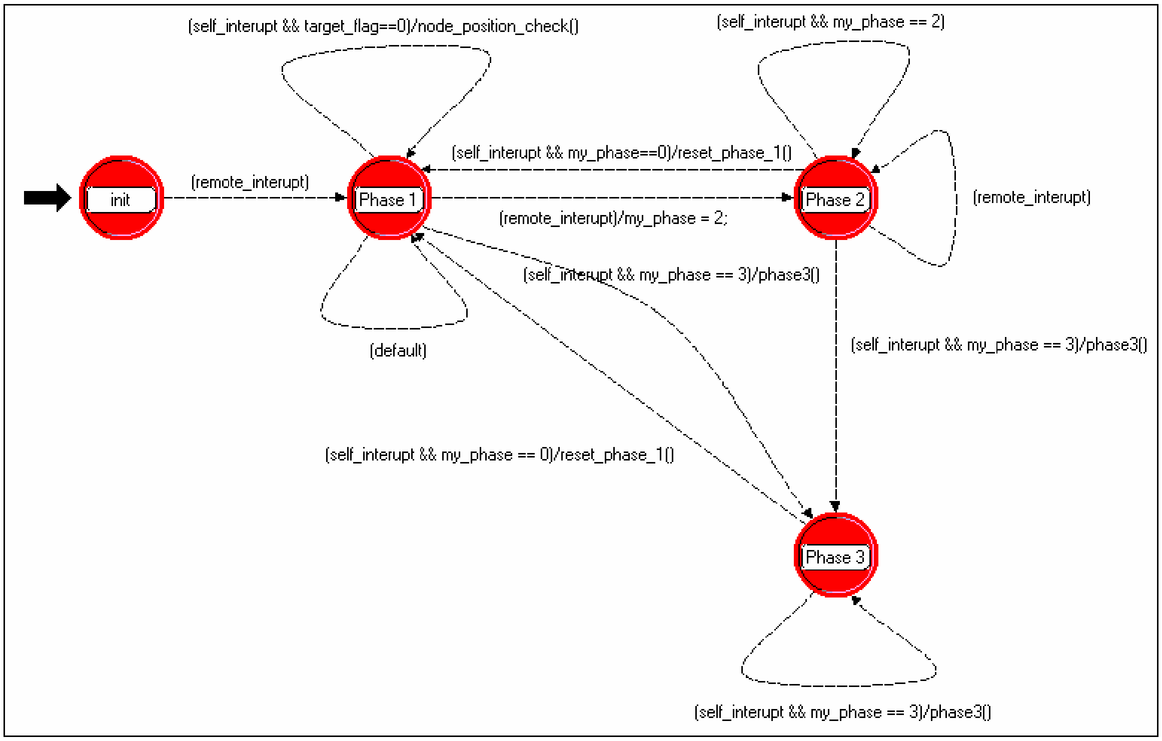

Figure 35. UAV Mobility CPU Process Model

During phase 1, the UAV checks every twentieth of a second (on average) to determine if the UAV has entered and is able to scan a new cell. If so, the UAV picks a new destination based upon the global search rule and broadcasts its next destination to the other UAVs in range. The time between successive position checks is the result of a uniform random process that produces results between 0.04 and 0.06 seconds. The reason that this is done is to keep the UAVs from being perfectly synchronized. When the UAVs are synchronized, it is possible that several of them will scan a new cell at the same instant and try to broadcast their next destinations. If the transmission medium has been idle long enough, each of them sends its packet immediately. This causes packets to collide at an inflated rate as compared to what would realistically happen, as it is unlikely that the UAVs will share the same master clock as they do in OPNET. When the UAV performs a position check and scans the cell, if a target is detected in its sensor radius, then the UAV will transition to phase 3. Otherwise, it will continue with the global search phase.

A UAV can only enter phase 2 from phase 1 if it receives a target packet. The decision to transition to phase 2 is made by the Mapping CPU and communicated to the Mobility CPU through a remote interrupt. While in phase 2, the UAV will approach the target. If the UAV enters and is able to scan cells along the way, it reacts the same way as it would in phase 1, except it broadcasts its current position because it does not choose a destination as it would in phase 1.

While in phase 2, the UAV also continues to listen to network traffic. If another target packet is received for the target, the UAV records this information and updates its target information table. If it is determined that three UAVs are orbiting the target the UAV in phase 2 will return to phase 1 and will set an ignore timer for the target. While the ignore timer has not expired, target information will still be stored in the target information table, but the UAV will not return to phase 2 and approach the same target until the ignore timer has expired. However, if there are three UAVs in orbit, it is likely that the UAV will not decide to enter phase 2 again for that target. While in phase 2 if a target packet is received and the UAV would have decided to approach that target, the UAV will evaluate whether the new target or the target it was approaching is closer and will approach the closer of the two targets. When a target, either a new target or the target the UAV was approaching is detected by the UAVs sensor, the UAV will enter phase 3 and orbit the target.

In phase 3, the UAV orbits the target and takes periodic measurements of the target's location and broadcasts this information as a target frame. Because the sensors used are assumed to only be able to detect the direction that the target is from the UAV, the UAV must obtain three suitable points for triangulation. It does not matter to the UAV whether the measurement taken was its own or was taken by another UAV orbiting the target. However, if the UAV determines that three different UAVs have made measurements and broadcast them more recently than its current measurement, the UAV sets an ignore timer for the target and returns to phase 1. If more than one UAV is orbiting a target, the UAVs attempt to spread out in their orbit in order to more quickly refine the target's location. For two UAVs in orbit, an ideal separation of ninety degrees is desired, while for three UAVs equal spacing between the UAVs is desired [YPH06].

When any UAV in orbit around a target determines that it has suitable information to detect the target, it performs several actions. The first of these is to send an interrupt to the target signaling that the target has been located. This causes the target to stop moving and transition to a disabled state. Next the UAV sends an interrupt to the *search configuration node*, described in Section A.6, so that proper animation and global statistical information is produced. Finally, the UAV sends a target packet that initiates a network flood so that all other UAVs are aware that the target has been located and disabled.

As stated before, the Mobility Support CPU has become nearly vestigial. The Mobility Support CPU's process model is presented in Figure 36. After initialization, the process creates periodic self interrupts to check that the UAV is still within the search boundaries and to produce regularly spaced animation frames. When the UAV reaches one of the area boundaries, the Mobility Support CPU causes it to move along the search area boundary. This is different than the case when a target reaches a boundary and "bounces" off the boundary as discussed in Section A.4. If the UAV is in phase 1, it will never cross the boundaries in correct operation. However, when approaching or orbiting a target, it is possible for the UAV to go beyond the boundary. It is not problematic if the UAV does go beyond the boundaries. This is because once the UAV finishes orbiting a target and returns to phase 1, the UAV will pick a destination within the search area boundaries. OPNET Modeler does not have problems handling nodes that go beyond the boundaries. However in the animation, the UAV's icon will stop at the boundary, but its actual x and y positions are still correctly maintained. The boundary check function was useful during development because it kept the UAVs in bounds. Though it currently does affect the simulation results, the function should be able to be removed with little consequence.

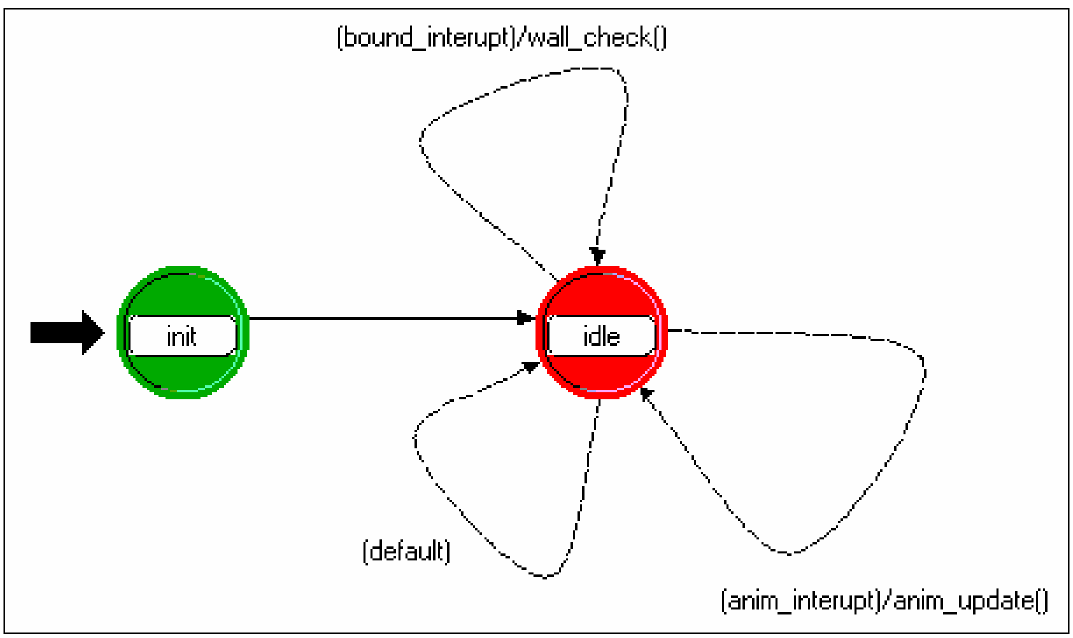

Figure 36. UAV Mobility Support CPU Process Model

Animation updates are created for an object when the op_ima_obj_pos_get() kernel procedure is called. Because the position of objects is important in the calculation of wireless network communications, this function is called often. During development this function was necessary when network traffic was not being generated in order to ensure that animation information was created for the purpose of debugging the simulation. Since network traffic is present in sufficient quantity to produce enough animation frames, this function is also no longer necessary.

A.6 Background Processing and Animation

The centralized statistics, simulation termination, and all animation functions are performed by the *search configuration node*. The icon for this particular node is the cloud shown in Figure 37. The *search configuration node* requires the user to provide several global attributes, which are necessary for the simulation to run. These include the search area boundaries, dimensions of individual cells, and whether to terminate the search when the last target is located or when the last cell is searched.

Figure 37. Search Configuration Node Icon

The *search configuration node*, like the target nodes, contains only a single processor; however, unlike the target nodes, this processor contains only a single process model, which is shown in Figure 38. During initialization, the *search configuration node* creates a map of the search area similar to that of the UAVs. However, instead of storing the last time a cell was searched, this map stores the number of times individual cells are searched. In addition to this map, the node draws a blue square that marks the central quarter area of each cell. The animation object handles are stored so that these blue squares can be erased later. The *search configuration node* also counts the number of targets and UAVs present during the initialization stage. A list of the UAVs' node identifiers is stored for quick reference of the sensor range animation function. The sensor range animation function is started during the initialization stage, but continually reschedules itself every five simulation seconds. When executed, this function draws a non-persistent yellow circular outline around each UAV to denote the greatest extent of the UAV's sensor range. Since sensor range is a global parameter, the sensor range of all UAVs is the same.

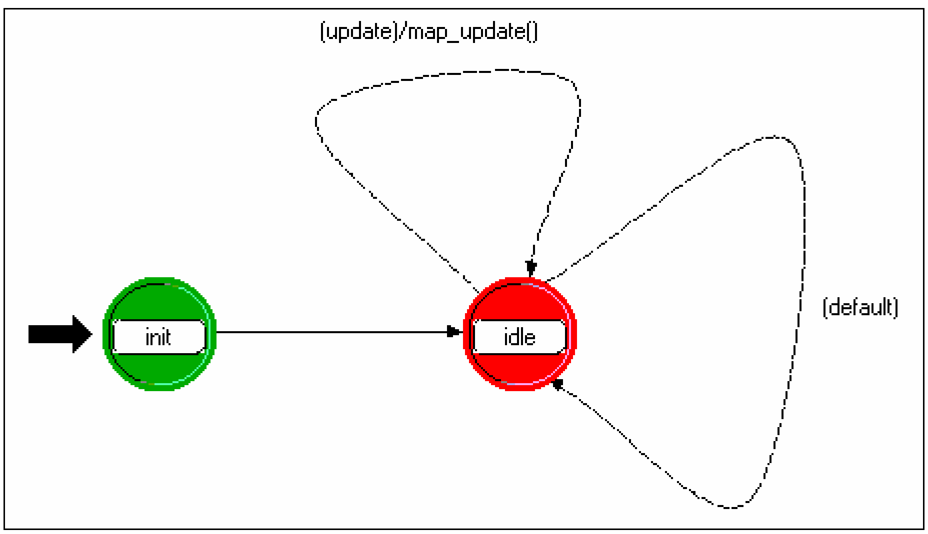

Figure 38. Search Configuration Node Process Model

Once the initialization stage is complete, the *search configuration node* process transitions to the idle state where it waits for an external node to provide and update interrupt rather than polling all of the targets and UAVs and determining what action must be taken. When a target changes its transmitting attribute, it generates an interrupt for the *search configuration node*. The *search configuration node* determines which state the target is in and updates its icon. The standard humvee icon is used when a target is transmitting, and a yellow circle is used when the target is not transmitting. When a UAV scans an area, it sends a packet to communicate this with other UAVs, but sends an interrupt to the *search configuration node*. When this occurs, the *search configuration node* increments the scan count for the cell the UAV just scanned and if the cell was scanned for the first time, the *search configuration node* erases the blue square in that cell. The last event that causes an interrupt to be generated for the *search configuration node* is when a target is located by a UAV. The UAV that locates the target first sends an interrupt to the target and then sends an interrupt to the *search configuration node*. This is important because the target must first be disabled so that the *search configuration node* will recognize that the target's icon needs to be changed to a red circle, designating a destroyed target, and that the number of remaining targets should be decremented.

Regardless of which reason an update interrupt is triggered, the *search configuration node* will update any necessary statistics. The *search configuration node* tracks the following statistics: total scans performed, number of cells scanned, percent of cells scanned, number of targets remaining, percent of targets found, and the search redundancy. All of the statistics listed, with the exception of search redundancy are recorded after every interrupt. The search redundancy statistic is only recorded when the simulation is terminated by the *search configuration node*.

The simulation terminates based upon the global Termination Condition parameter. This parameter can be configured so that the search will terminate when all of the targets have been located or when every cell has been scanned at least once. If neither of these events happens before the maximum simulation duration is reached, then the simulation will be terminated, but the search redundancy statistic is not recorded. For normal operation, the duration of the simulation should be set long enough to allow the simulation to be terminated by the *search configuration node*.

Figure 39 provides a screenshot of an example simulation. The parameters in the example are not configured as discussed in the main body of this document. The cell dimensions were set at 20 meters and the background grid shows the cell boundaries. The UAVs have a sensor radius of 25 meters. The screenshot was taken at a point where all possible symbols are shown with as much clarity as possible (some of the trajectories are hidden by the blue squares). Table 12 provides a summary of the symbols used in the animation and serves as a legend for the screenshot.

Table 12. Animation Symbol Legend

Object	Animation Symbol
UAV	Plane
Target (Transmitting)	Humvee
Target (Not Transmitting)	Filled Yellow Circle
Search Configuration Node	Cloud
UAV or Target's Current Bearing	Green Arrow
Unscanned Cell	Blue Square in central quarter area
Destroyed / Located Target	Filled Red Circle
UAV Sensor Detection Limit	Yellow Circular Outline around UAV

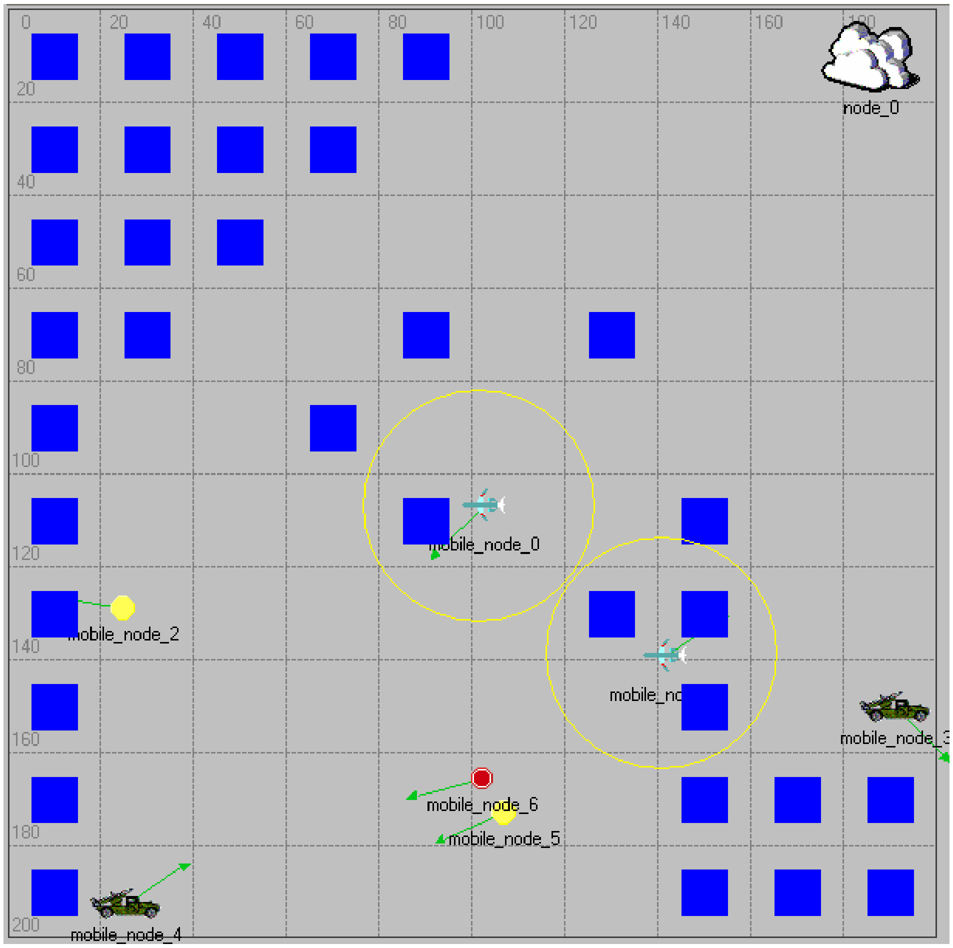

Figure 39. Sample Screenshot of OPNET Animation

Appendix B – Additional Validation Plots

This appendix provides the plots omitted in Section 4.1 that support the validation of the communication network used in the simulations. Some subfigure legends have been omitted so that the sizes of the figures could be made larger; all related subfigures share the same legend. The solid lines in each figure represent the data taken from generic network nodes. The dotted lines, using the same color, show the data collected from UAVs when targets were not present in the models.

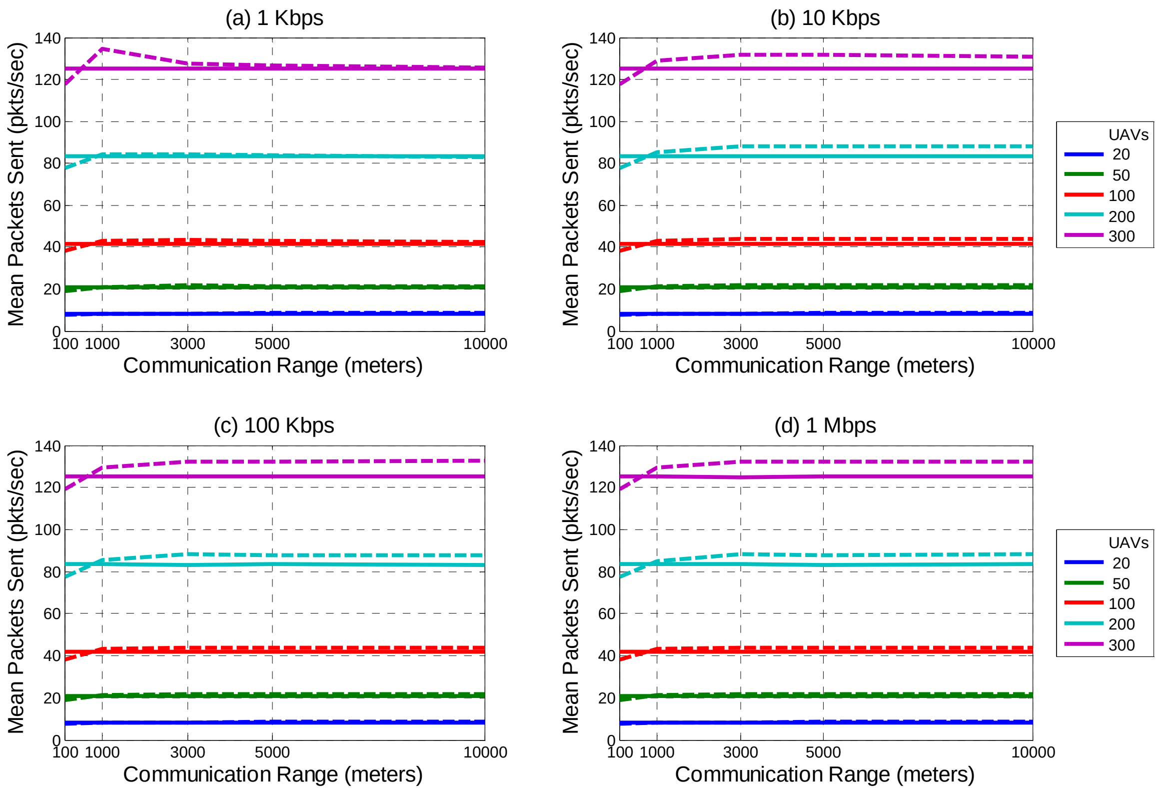

Figure 40. Mean Packets Sent without Targets versus Communication Range

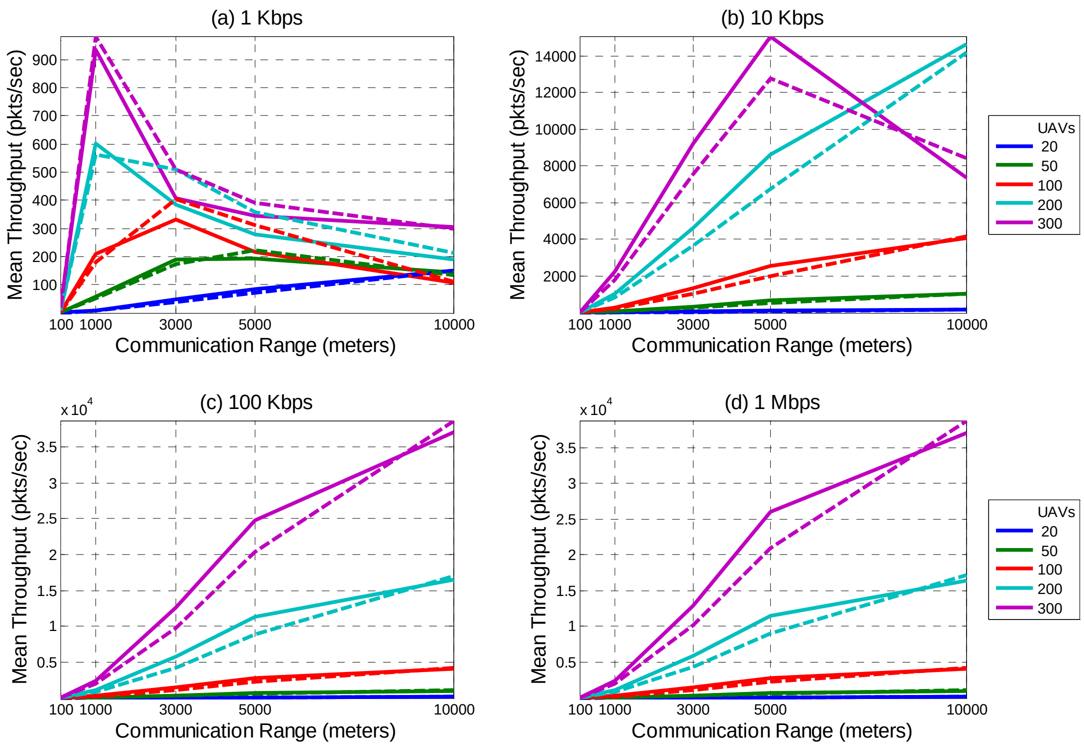

Figure 41. Mean Throughput without Targets versus Communication Range

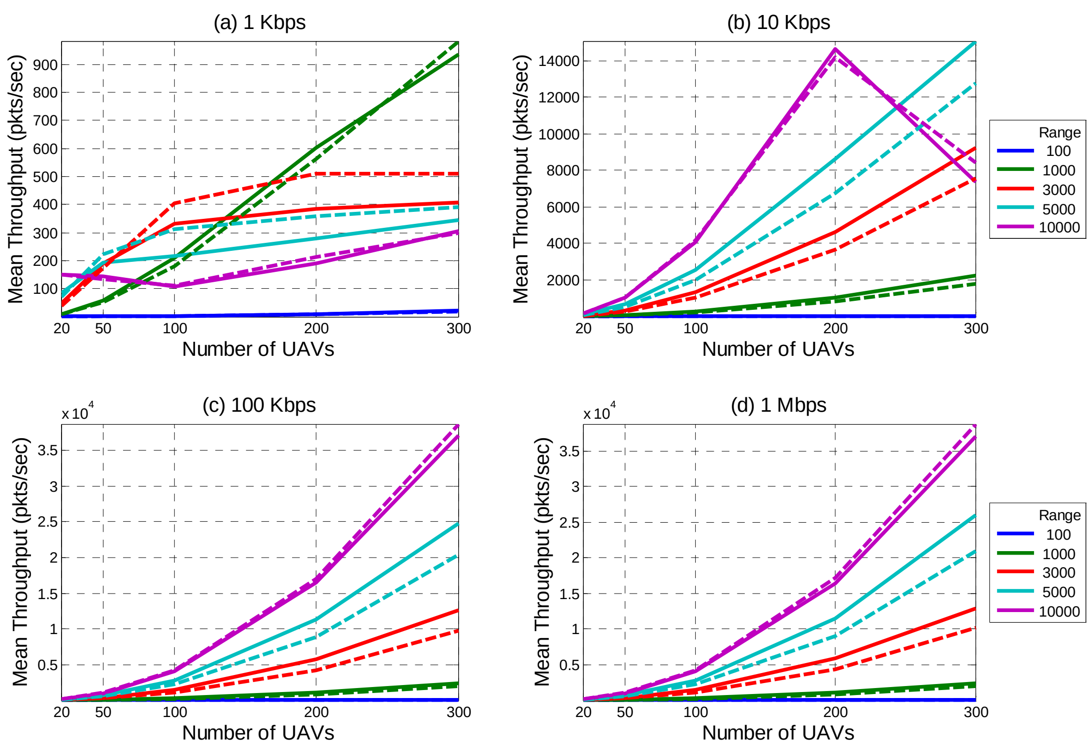

Figure 42. Mean Throughput without Targets versus the Number of UAVs

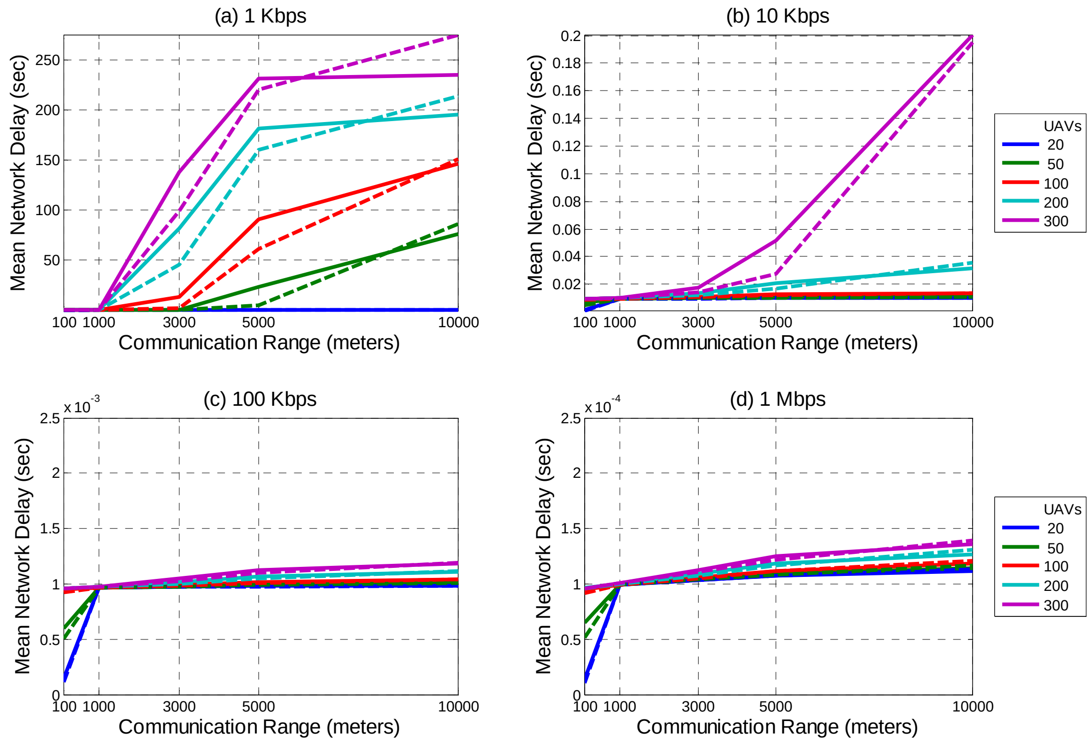

Figure 43. Mean Network Delay without Targets versus Communication Range

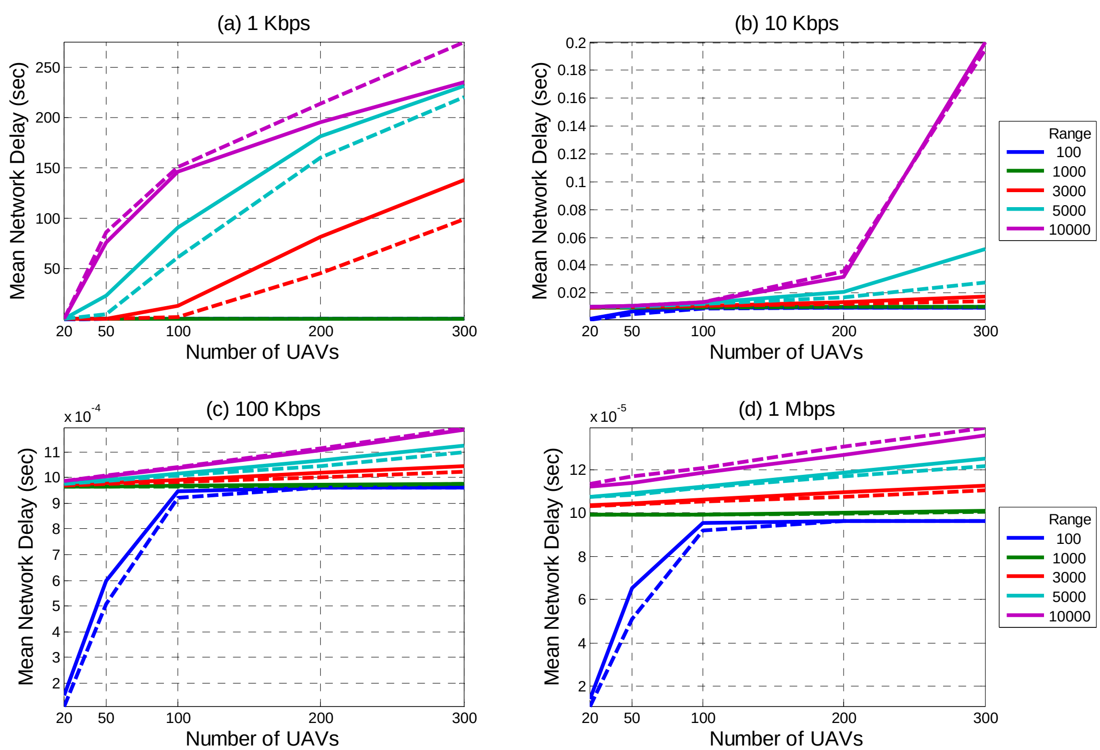

Figure 44. Mean Network Delay without Targets versus Number of UAVs

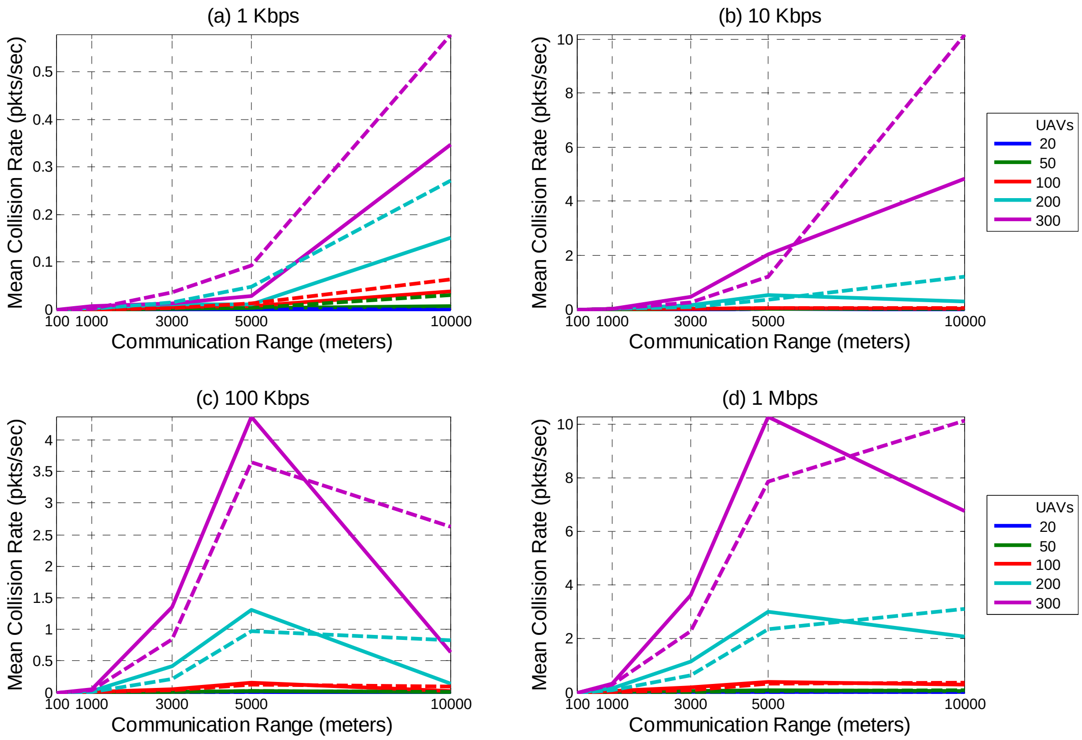

Figure 45. Mean Collision Rate without Targets versus Communication Range

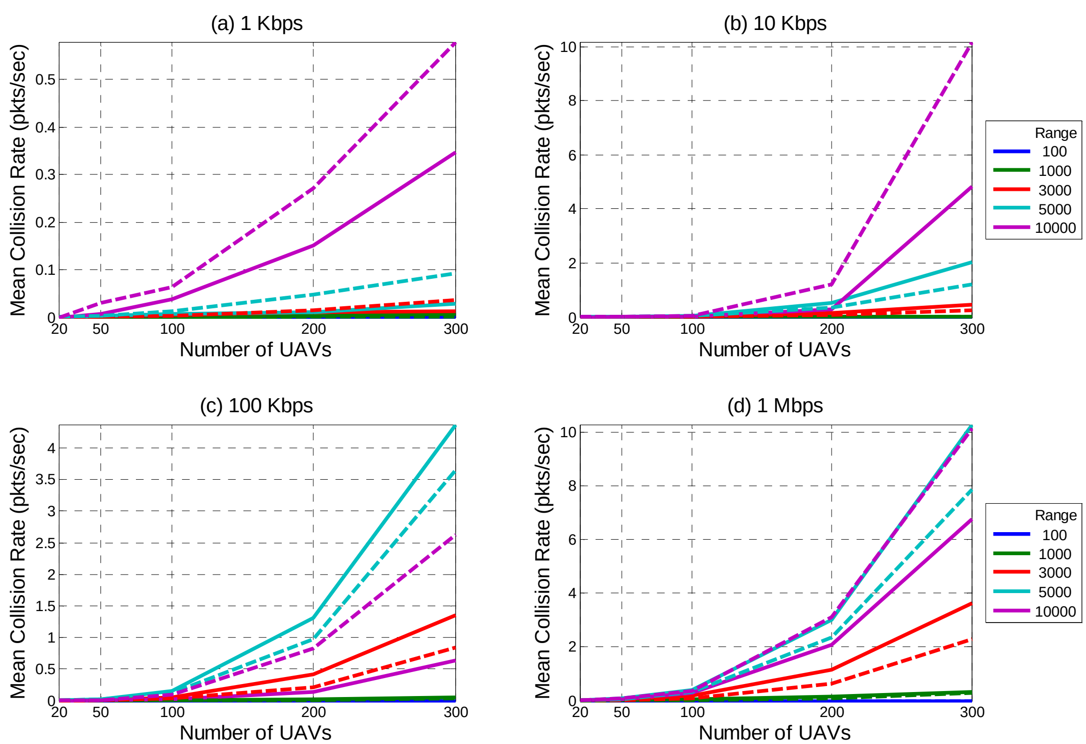

Figure 46. Mean Collision Rate without Targets versus Number of UAVs

Appendix C – Additional Network Performance Plots

The plots in this appendix are meant to augment those presented in Section 4.2.3. These plots present a different view of the data collected. Specifically, the data is plotted against the number of UAVs rather than the communication range.

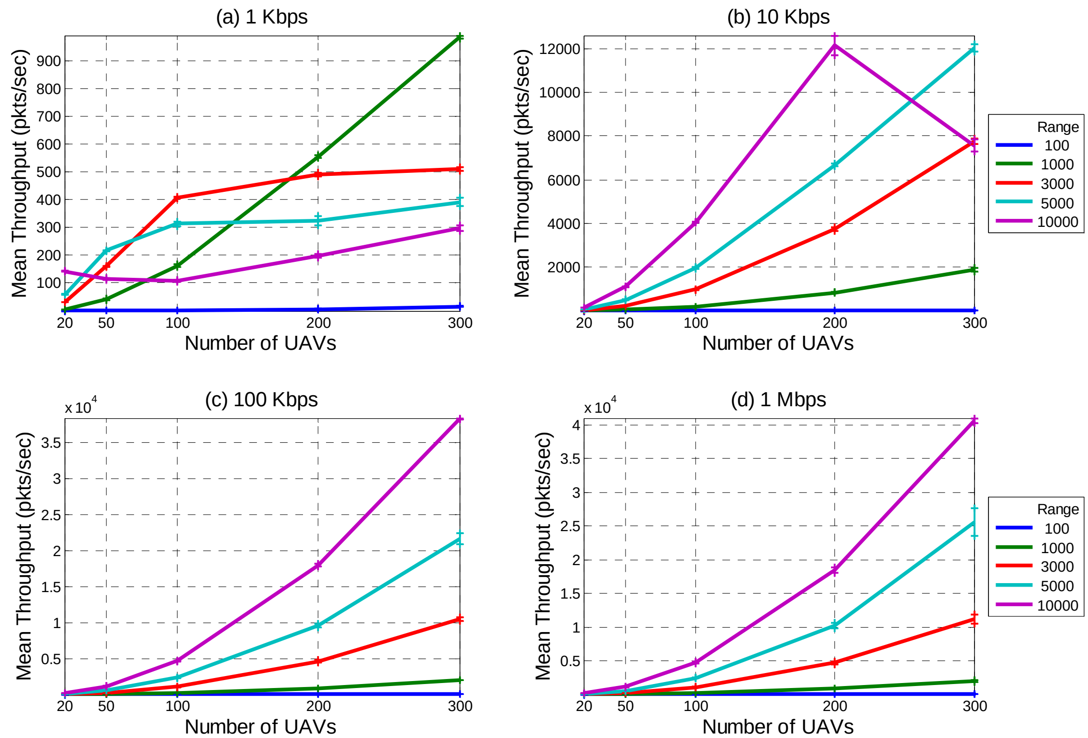

Figure 47. Mean Throughput versus Number of UAVs

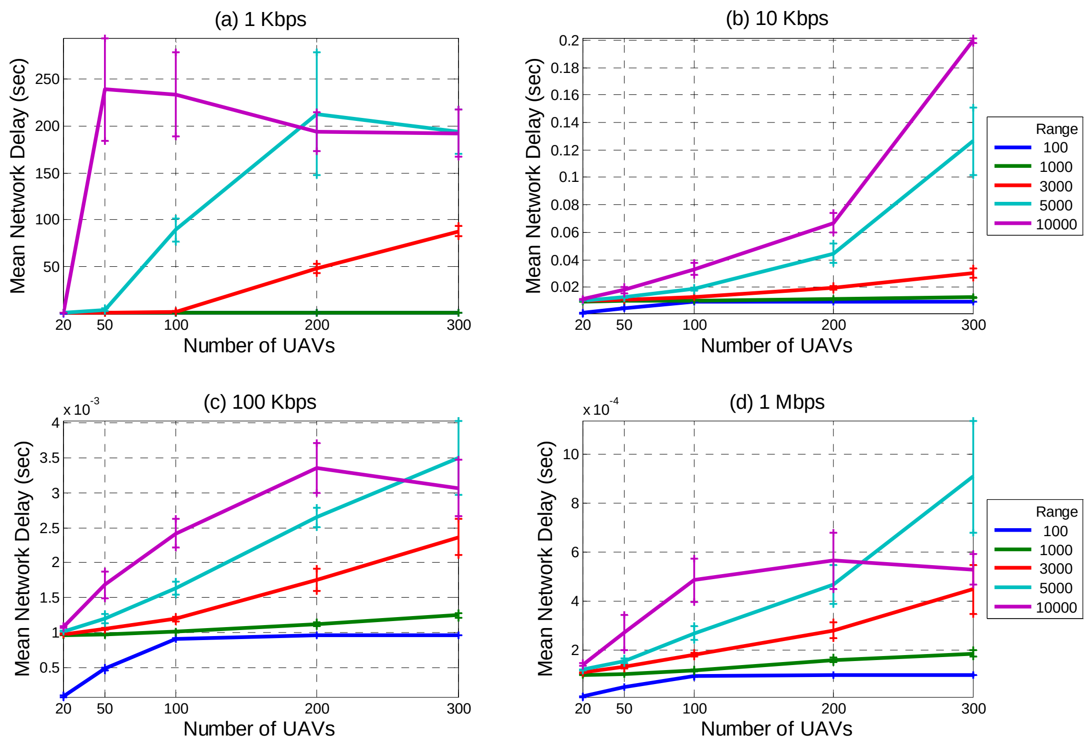

Figure 48. Mean Network Delay versus Number of UAVs

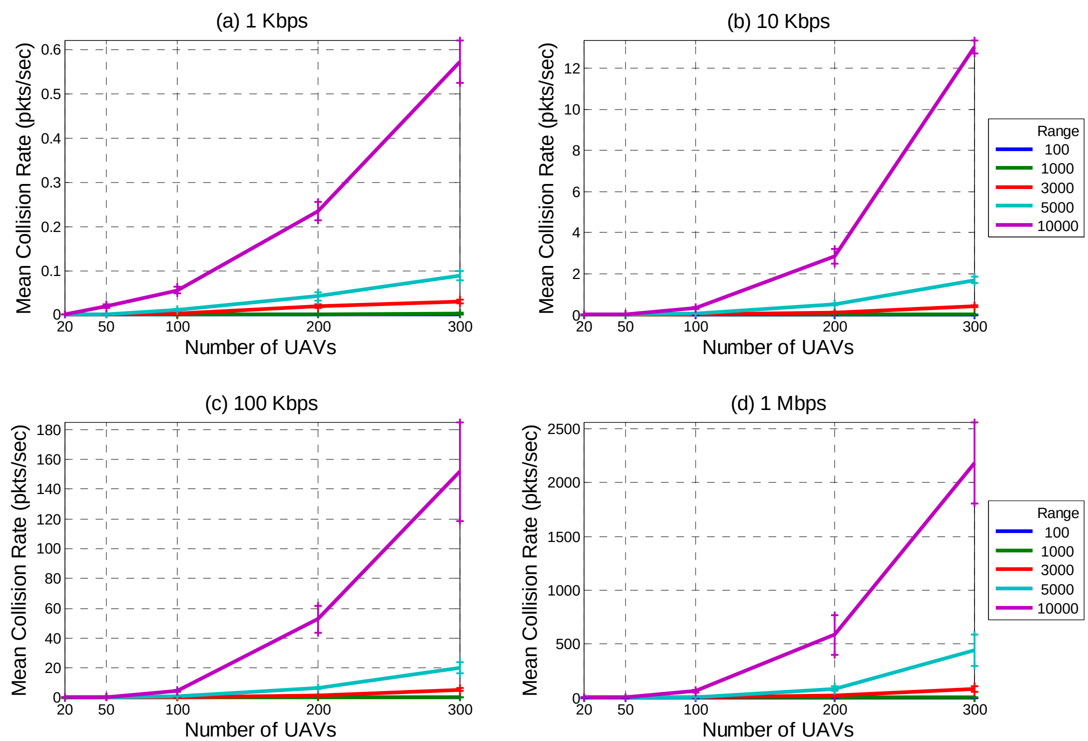

Figure 49. Mean Collision Rate versus Number of UAVs

Bibliography

[BDT99] Eric Bonabeau, Marco Dorigo, and Guy Theraulaz. *Swarm Intelligence: From Natural to Artificial Systems*. New York: Oxford University Press, 1999.

[Bia00] Giuseppe Bianchi, "Performance Analysis of the IEEE 802.11 Distributed Coordination Function" in *IEEE Journal on Selected Areas in Communications*, Vol. 18, No. 3, March 2000, pp. 535-547.

[Bre97] Pablo Brenner, "A Technical Tutorial on the IEEE 802.11 Protocol", Breezecom Wireless Communications Publication, 1997. Available online at: http://www.sss-mag.com/pdf/802_11tut.pdf.

[CBD02] Tracy Camp, Jeff Boleng, and Vanessa Davies, "A survey of mobility models for ad hoc network research" in *Wireless Communications and Mobile Computing*, Vol. 2, 2002, pp. 483-502.

[CVB02] P. Chatzimisios, V. Vitsas, and A.C. Boucouvalas, "Throughput and delay analysis of IEEE 802.11 protocol", *Proceedings of the IEEE 5th International Workshop on Networked Appliances*, Liverpool, October 2002, pp. 168-174.

[DoD01] Department of Defense (DoD). *Department of Defense Dictionary of Military and Associated Terms*. Joint Pub 1-02. Washington: HQ DoD, April 2001 (as amended through 30 November 2004).

[Gar02] Jim Garamone, "Unmanned Aerial Vehicles Proving Their Worth Over Afghanistan", American Forces Press Service, 16 April 2002. Available online at *Defenselink*: http://www.defenselink.mil/news/Apr2002/n04162002_200204162.html

[Gle99] Russell W. Glenn, *"… We Band of Brothers": The Call for Joint Urban Operations Doctrine*, Santa Monica, California: RAND Corporation, 1999.

[Hob05] William T. Hobbins, General, Warfighting Integration and Chief Information Officer of the Air Force (SAF/XC), Address to Air Force Institute of Technology students and faculty. Air Force Institute of Technology, Wright-Patterson AFB OH, 24 Oct 05.

[HoC96] Tien-Shin Ho and Kwang-Cheng Chen, "Performance analysis of IEEE 802.11 CSMA/CA medium access control protocol" in the *Seventh IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*, Vol. 2, October 1996, pp. 407-411.

[Jan05] Jane’s Information Group, “Jane’s Unmanned Aerial Vehicles and Targets”, http://juav.janes.com, 1 March 05.

[Jon05] Tina W. Jonas, Testimony on the FY 2006 Department of Defense Budget, Senate Appropriations Committee Subcommittee on Defense, 2 March 2005. Available online at: http://appropriations.senate.gov/subcommittees/record.cfm?id=232624

[Kad03] Brian Kadrovach, “A Communications Modeling System for Swarm-based Sensors,” Ph.D. dissertation, Dept. of Elect. and Comp. Eng., Air Force Institute of Technology (AU), Wright-Patterson AFB, OH, 2003.

[KEM01] I. Kassabalidis, M. A. El-Sharkawi, R. J. Marks, P. Arabshahi, and A. A. Gray, “Swarm Intelligence for Routing in Communication Networks,” in *Global Telecommunications Conference, 2001*, IEEE, San Antonio, Texas, November 2001, Vol. 6, pp 3613-3617.

[LAN03] Chin A. Lua, Karl Altenburg, and Kendall E. Nygard, “Synchronized Multi-Point Attack by Autonomous Reactive Vehicles with Simple Local Communication,” in *Proceedings of the 2003 IEEE Swarm Intelligence Symposium,* April 2003, pp. 95-102.

[LHP05] Jae-Min Lee, Soo Hee Han, Hong Seong Park, and Wook Hyun Kwon, "Performance Analysis of the IEEE 802.11 DCF with Time-Varying Channel Environments," *IEICE Transactions on Communications,* Vol. E88–B, No. 9, September 2005, pp. 3784-3787.

[MeW06] Merriam-Webster Online Dictionary, “Definition of swarm”, http://www.m-w.com/dictionary/swarm, 4 January 06.

[MMP06] Kevin M. Morris, Barry E. Mullins, Daniel J. Pack, George W.P. York, and Rusty O. Baldwin, “Impact of Limited Communications on a Cooperative Search Algorithm for Multiple UAVs,” *Proceedings of 2006 IEEE International Conference on Networking, Sensing and Control*, to be published April 2006.

[New04] Laurence R. Newcome, *Unmanned Aviation: A Brief History of Unmanned Aerial Vehicles*, Virginia: American Institute of Aeronautics and Astronautics, Inc., 2004.

[Opn04] OPNET Product Documentation Team, “Modeling Wireless Networks,” *Wireless Module Users Guid*e, OPNET Modeler 10.5.A Product Documentation, 2004.

[PaE99] Julia K. Parrish and Leah Edelstein-Keshet, “Complexity, Pattern, and Evolutionary Trade-Offs in Animal Aggregation,” in *Science*, Vol. 284, 2 April 1999, pp. 99–101.

[PaM03] Daniel J. Pack and Barry E. Mullins, “Toward Finding an Universal Search Algorithm for Swarm Robots,” in the *Proceedings 2003 IEEE/RJS Conference on Intelligent Robotic Systems (IROS)*, Las Vegas, Nevada, October 2003.

[PeD03] Larry L. Peterson and Bruce S. Davie, *Computer Networks: A Systems Approach, Third Edition*. New York: Morgan Kaufmann Publishers, 2003.

[PPO02] H. V. D. Parunak, M. Purcell, and R. O’Connell, “Digital Pheromones for autonomous coordination of swarming UAVs,” in *Proceedings of the AIAA's First Technical Conference and Workshop on Unmanned Aerospace Vehicles, Systems, and Operations.* Portsmouth, Virginia, May 2002.

[PVG02] Julia K. Parrish, Steven V. Viscido, Daniel Grünbaum, “Self-Organized Fish Schools: An Examination of Emergent Properties,” *Biological Bulletin*, Vol. 202, June 2002, pp. 296-305.

[PYT05] Daniel Pack, George York, and Gregory Toussaint, “Localizing Mobile RF Targets using Multiple Unmanned Aerial Vehicles with Heterogeneous Sensing Capabilities,” *2005 IEEE International Conference On Networking, Sensing and Control* in Tucson, Arizona, 19-22 March 2005.

[Ric05] Amber J. Rickmon, “Evaluation of the Ad Hoc On-demand Distance Vector Routing Protocol for Mobile Ad Hoc Networks,” Masters Thesis, Dept. of Elect. and Comp. Eng., Air Force Institute of Technology, Wright-Patterson AFB, OH, 2005.

[RoT99] Elizabeth M. Royer and Chai-Keong Toh, “A Review of Current Routing Protocols for Ad Hoc Mobile Wireless Networks,” in *IEEE Personal Communications*, April 1999.

[Sam03] Doug Sample, “Pentagon Plans Heavy Investment in UAV Development”, American Forces Press Service, 18 March 2003. Available online at: http://www.iwar.org.uk/news-archive/2003/03-18-2.htm

[Swe99] Craig Sweet, “Performance Analysis of the IEEE 802.11 Wireless LAN Standard,” Masters Thesis, Department of Computer Science and Electrical Engineering, University of Maryland Graduate School, Baltimore, MD, 1999.

[TaK85] Hideaki Takagi and Leonard Kleinrock, "Throughput Analysis for Persistent CSMA Systems," in *IEEE Transactions on Communications*, Vol. COM-33, No. 7, July 1985, pp. 627-638.

[Thu02] Charles N. Thurwachter, Jr., *Wireless Networking*, Columbus, Ohio: Prentice Hall, 2002.

[ViR04] Patrick Vincent and Izhak Rubin, “A Swarm-assisted Integrated Communication and Sensing Network” in *Proceedings of SPIE*, Vol. 5441, 2004, pp. 48-60.

[Wik05] Wikipedia, “Dynamic Source Routing”, http://en.wikipedia.org/wiki/Dynamic_Source_Routing, 22 June 05.

[Wik06] Wikipedia, “Emergence”, http://en.wikipedia.org/wiki/Emergence, 4 January 06.

[YPH06] George W. P. York, Daniel J. Pack, and Jens Harder, “Comparison of Cooperative Search Algorithms for Mobile RF Targets using Multiple Unmanned Aerial Vehicles,” invited chapter in *Cooperative Control and Optimization*, Gainesville, Florida: University of Florida Press, 2006.

Vita

Kevin Morris was born in Midland, Texas, in 1982 and graduated from Robert E. Lee High School in 2000. He was commissioned as a Second Lieutenant through the Air Force Reserve Officer Training Corps (AFROTC) at Texas Tech University in May 2004. At Texas Tech, he completed a dual degree program and earned Bachelor of Science degrees in Electrical Engineering and Computer Science. He received the Texas Tech's President's Award for being the top AFROTC graduate in his class as well as the Air Force Communications and Electronics Association's (AFCEA) Maj. Gen. Robert E. Sadler Honor Award as the top Electrical Engineering graduate from AFROTC for that year. Lt Morris was selected to attend the Air Force Institute of Technology (AFIT) and reported to Wright Patterson AFB, OH, in August 2004, after first attending the Air and Space Basic Course (ASBC) at Maxwell AFB, AL (Class 04E). Lt Morris will be assigned to the Emerging Technologies Division of the Air Force Communications Agency at Scott AFB, IL, upon graduation from AFIT.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 074-0188

1. REPORT DATE *(DD-MM-YYYY)*	**2. REPORT TYPE**	**3. DATES COVERED** *(From – To)*
23-03-2006	Master's Thesis	March 2005 – March 2006

4. TITLE AND SUBTITLE

Performance Analysis of a Cooperative Search Algorithm for Multiple Unmanned Aerial Vehicles Under Limited Communication Conditions

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

Morris, Kevin M., Second Lieutenant, USAF

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAMES(S) AND ADDRESS(S)

Air Force Institute of Technology
Graduate School of Engineering and Management (AFIT/EN)
2950 Hobson Way, Building 640
WPAFB OH 45433-8865

8. PERFORMING ORGANIZATION REPORT NUMBER

AFIT/GE/ENG/06-44

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

United States Air Force Academy, UAV Research Group
Contacts: Daniel Pack, Ph.D. and Lt Col George W.P. York, Ph.D.
Mailing Address: USAFA/DFEC, 2354 Fairchild Drive, Suite 2F6
USAF Academy, Colorado Springs, CO 80840
Phone: (719) 333-6967 DSN: 333-6967 Fax: (719) 333-3756

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

This research investigates the impacts of realistic wireless communications upon a group of unmanned aerial vehicles (UAVs) utilizing a distributed search algorithm. The UAVs are used to survey an area for mobile targets and require communication to cooperatively locate the targets. The mobile targets do not continually radiate energy, which exacerbates the search effort; a UAV could fly directly over a target and not detect it. A simulation of cooperative UAVs is implemented using the OPNET Modeler network simulation tool.

The search performance of a group of UAVs is observed when (1) communication range, (2) data rate, and (3) the number of UAVs are varied. The performance is evaluated based on the total time it takes for the UAVs to completely detect all targets in a given search area, the number of times internal areas are scanned, as well as the amount of communication throughput achieved, network traffic generated, network latency, and number of network collisions. The results indicate that the number of UAVs was found to have the greatest impact on the group's ability to search an area, implying the data shared between UAVs provides little benefit to the search algorithm. In addition it is found that a network with a 100 Kbps or faster data rate should allow for minimal congestion and for a large degree of scalability.

15. SUBJECT TERMS

Unmanned Aerial Vehicle (UAV), cooperative, distributed search algorithm, sensor swarm, swarm, wireless communication, IEEE 802.11b.

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON**
a. REPORT	**b. ABSTRACT**	**c. THIS PAGE**			Barry E. Mullins, Ph.D. (ENG)
					19b. TELEPHONE NUMBER *(Include area code)*
U	U	U	UU	138	(937) 255-3636, ext 7979 (Barry.Mullins@afit.edu)